DATA HIDING SCHEME WITH GEOMETRIC DISTORTION CORRECTION

XIAOXIA JIANG

DATA HIDING SCHEME WITH GEOMETRIC DISTORTION CORRECTION

by

©Xiaoxia Jiang

A thesis submitted to the
School of Graduate Studies
in partial fulfillment of the
requirement for the degree of
Master of Science

Department of Computer Science
Memorial University of Newfoundland

December 2007

St. John's Newfoundland Canada

Acknowledgement

First of all, I would like to thank my parents for their care, support and advice.

It is a great pleasure to thank my supervisor, Dr Siwei Lu, who has always been both, critical and supportive. Without his help and suggestions, this would be less organized, harder to understand and contain more errors, definitely.

I want to thank all my colleagues and local technical staff for their patience. I am also very grateful to the administrative staff who have helped in one to another in the preparation of this thesis.

Furthermore, I would like to thank the principal authors of and all contributors to many open-source projects- software which greatly facilitated the task of writing this thesis.

Abstract

This work presents a new data hiding scheme with geometric distortion corrections. In a data hiding scheme, robustness of embedded information against geometric distortion is a critical issue, since a decoder can be easily confused when image size or orientation is changed, even to a small extent. One of the approaches to overcome this problem is to rescale and/or rotate the image back to its original size/orientation. In order to do so, the decoder needs to know by how much the image has been scaled and rotated. Such knowledge of the original image would greatly simplify the extraction of embedded information. However, huge quantities of images have to be compared by intelligent agents to locate the original images. Therefore, there is a strong practical reason to seek robust extraction schemes without access the original images.

In this approach, two separate channels of information are embedded in a typical RGB image by selecting two of its three color planes as two channels: one for synchronization information and the other one for the actual information. Synchronization information of the original image is computed by the locations of feature points extracted by a Harris-Laplace corner detector and embedded as a watermark into the image by a content-based watermarking method. This synchronization information is extracted at the detection stage and compared with the calculated synchronization information of the distorted image so that the scaling factor and rotation angle can be estimated.

The effectiveness of the presented scheme is evaluated by assessing its robustness against different geometric attacks and transformations. The geometric distortion

estimation shows best robustness in certain ranges of scaling and rotation distortions. The proposed scheme exhibits great potential because of the successful detection of synchronization information.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Watermarking techniques have recently received considerable attention from the research community and from industry. The main driving force for watermarking is the concern with copyright protection in music, film, book and software publishing industries. As audio, video and other works become available in digital form, perfect copies can be made easily. Moreover, the growth of the Internet may lead to large-scale unauthorized copying. 'Copyright marking' techniques that identify the copyright holder of the work by embedding a "mark" into host or cover data have been well established. Currently, there are copyright marking methods for virtually every kind of digital media: text documents [1, 2], images [3], video [4, 5], audio [6, 7], and even 3D polygonal models [8].

Other than copyright protection, another important application of watermarking techniques is steganography. Steganography literally means 'covered writing', and is usually interpreted as hiding information in other information. Steganography is not a new science subject. It has been widely used over the centuries for analog media but today is being applied for digital multimedia contents. Examples include sending a message to a spy by marking certain letters in a newspaper using invisible ink, and adding sub-perceptible echo at certain places in an audio recording. Steganography techniques attract much attention in military communications. Rather than encrypting the message, data hiding is used to hide the very existence of the message. This allows communication using often enciphered messages without attracting the attention of a third party.

The general model of watermarking systems can be described as follows. A digital watermarking system consists of two main components: *watermark embedder* and *watermark detector*. The embedder combines an original copy of digital media, called *cover image*, and a collection of bits representing metadata to be added to the cover image, named *payload* and creates the *watermarked cover image*. The watermarked cover image is perceptually identical to the cover image but with the payload embedded within. The difference between the cover image and the watermarked image is referred to as *embedding distortion*. The payload is not directly added to the original cover image. Instead, it is first encoded as a *watermark*, possibly using a secret key. The watermark is then modulated / scaled, yielding a *modulated watermark*, to make the embedding distortion small enough or even imperceptible.

The watermarked cover image may be subjected to different types of processing prior to detection, yielding a corrupted watermarked cover image. This corruption,

whether intentional or incidental, is known as an *attack*. The difference between the cover image and the watermarked image is referred to as *noise*. A watermark detector either extracts the payload from the corrupted watermarked cover image, or it produces some types of confidence measures indicating how likely it is for a given payload to be present. The extraction of the payload is done with help of a watermark key.

Three conflicting aspects are usually used to describe the requirements of watermarking systems. The *robustness* is identified with the probability of decoding error or resistance against watermark attacks. The *capacity* is the maximum payload that the watermarking can reliably embed and retrieve. The *imperceptibility* of watermarking systems is that the watermark embedding process should not introduce any perceptible distortion into the cover image. The tradeoffs among these three requirements are used to characterize information hiding schemes. The purposes of various watermarking applications determine their different requirements.

Copyright watermarking aims to identify the copyright holder of work. It should be robustly extracted even after undergoing various accidental and malicious attacks. In other words, copyright marking is designed for robustness. Many watermarking schemes that handle different attacks have been introduced. However, there is a tradeoff between capacity and robustness in copyright marking. The embedded information is usually only one binary bit (yes or no), i.e. whether marks have been embedded into cover images or not. Unlike copyright watermarking, the purpose of data hiding is to invisibly embed the maximum amount of data into a cover image. Typically, robustness requirements are low for data hiding purposes. Instead, invisibility and capacity are of prime importance. Capacity is an important property because it has a direct negative impact on watermark robustness. Higher capacity (the amount of information being embedded) causes lower

watermark robustness. Usually it is assumed that there are not many robustness requirements in data hiding schemes, except trivial operations such as rotation, scaling, and translation which are hard to avoid in common image manipulations.

Attacks can be classified as signal processing attacks and geometric attacks. Much emphasis has been placed on the robustness of common signal processing operations. However, it has become clear that even very small geometric distortions can prevent the detection of a watermark. The geometric attacks can be viewed as losing synchronization problems in communication theory. Synchronization is the process of identifying the coordinates of an embedded watermark. If the detector's input is watermarked but synchronization fails, then the embedded watermark will not be detected. Therefore, even a slight geometric modification can defeat many existing watermarking algorithms.

In the first generation of watermarking scheme against geometric distortions, digital watermarking schemes use pixels, frequency or other transform coefficients to embed the information. In the second generation, image features such as the corner and feature points are used as an indicator to retrieve the original location of the embedded information.

Among the existing second generation algorithms, Masoud presented a method to estimate the scaling factor of a previously scaled watermarked image and the angle by which the image was rotated [9]. The method estimated the image scaling factor and rotated angle through how much the feature points scaled and rotated. Therefore, it only needs to compare the feature point information of the original image and the distorted image. Unfortunately, the detector usually does not have prior information of the original image. The purpose of this work is to investigate the possibility that the data hiding

scheme carries not only the large amount information needed to be hidden but also the prior information of feature points to estimate the geometric distortion.

1.1 System Overview

Two color spaces of RGB images can be considered as two independent channels [10]. One is the synchronization channel (SC) which transmits the prior information of the original image and the other is the communication channel (CC) carrying the hiding data.

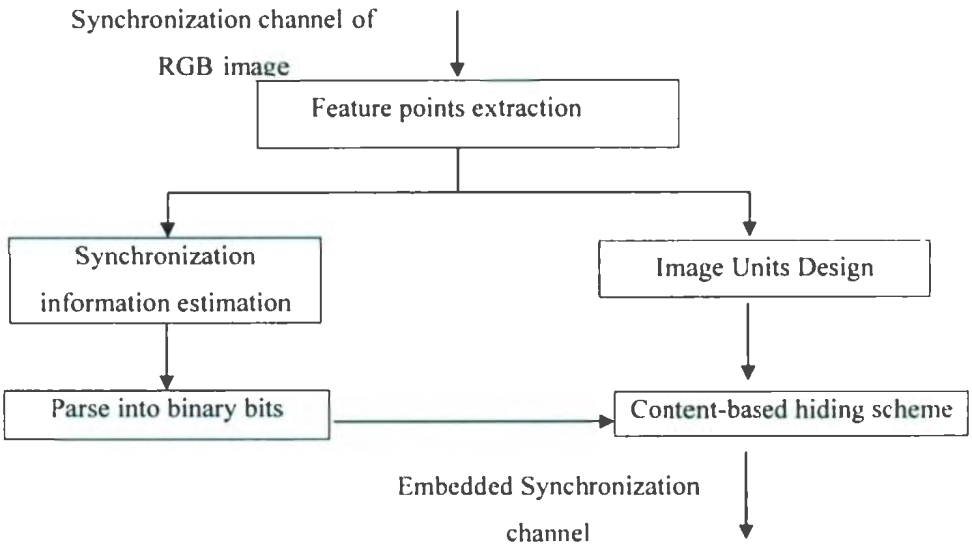


Figure 1.1.1 : Overview of synchronization channel hiding scheme

An overview of the proposed embedding scheme for the synchronization channel is shown in Figure 1.1.1. At the first stage, feature points are extracted and are used to obtain both synchronization information of the original image and construction of image units. A content-based watermarking scheme is applied in the synchronization channel to

fulfill the requirement of robustness against geometric distortions. At the last stage, the synchronization information of the original image is extracted and that of the actual image (distorted image) is calculated. The rescaling factor and rotation angle can be estimated by comparing these two sets of synchronization information.

1.2 Organization of Thesis

Chapter 2 clarifies applications of watermarking based on different requirements, refines the general information hiding model, and investigates the methods of information hiding and basic attacks.

Chapter 3 reviews the existing watermarking methods that are resistant to geometric attacks with a concentration on the content-based watermarking scheme since this scheme seems to be an excellent choice for improving the robustness against geometric distortions.

Chapter 4 presents a new scheme for data hiding with robustness against scaling and rotation distortions. The motivation and the whole watermarking scheme are illustrated. A few key problems such as improvement of the feature points extractor, image unit design and synchronization information estimation are presented.

Chapter 5 describes the performance of the scheme and discusses the capability. The precision of the feature point extractor and the capability of the scheme are further investigated.

Chapter 2

Survey of Watermarking Systems

Watermarking techniques can be classified by different criteria, including embedding domains [11], use of communications/information analysis [12] and inclusion of watermarking security [13]. This chapter surveys common embedding methods, watermarking schemes in transform domains and attack resistance methods.

2.1 Embedding Methods

Based on the watermark embedding/merging mechanism, most watermarking systems can be simply classified into two groups: those by linear addition of a spread spectrum signal and those by non-linear quantization-and-replace strategy, as described in the following sections.

2.1.1 Linear Additive of Spread Spectrum Signal

Additive embedding strategies are characterized by linear modification of the host image and correlative processing in the detection stage.

2.1.1.1 Concept of Spread Spectrum

The most common watermark requirements are invisibility and robustness. These requirements are somewhat contradictory to each other. Good invisibility suggests a low embedding strength/energy of the watermark signal, which can be considered as noise, in the media to avoid perception, whereas robustness requires high embedding strength/energy to help statistical detection. Therefore, an effective scheme of embedding compromising both invisibility and robustness is needed. Among many techniques proposed, spread-spectrum communication theory is the most commonly used and has proven a good solution for the invisibility-robustness contradiction.

Using spread-spectrum communication theory, watermark signals spread in low amplitude but in a wide enough bandwidth to hold enough signal embedding

strength/energy for detection. Many watermarking techniques incorporate the idea of spread spectrum communication to additively embed and extract a pseudo-random noise pattern [14-19]. The information bits spread by simple repetition [14], error-corrective coding [20], or some other transforms and then are modulated with a cryptographically secure pseudo-random noise sequence. The sequence embedded in the cover image can be either a Gaussian noise, a binary data, or a small image (a "logo"). The spread watermark signal is similar to the noise present in images and therefore is hard to detect [21].

2.1.1.2 Correlation Detector

For most additive watermarking methods, watermark detection is based on computing the linear correlation between the transmitted watermark and the received image and comparing the correlation value to a threshold. If the calculated linear correlation is small, then a conclusion can be made that the image is not watermarked. Otherwise, the image was watermarked. This decision is usually made based on a threshold. Thus the choice of the threshold influences the probability of false-positive error and false-negative error. The false-positive error represents the fact that un-marked images are detected as marked images, and false-negative error indicates that watermarked images are detected as un-marked images. Hence, a lot of effort has been used to devise reliable methods to compute predictable correlation thresholds and efficient watermark detection systems [22, 23]. Using two separate pseudo-random noise patterns could be a solution to conceal the requirement of setting a threshold [15]. This increases the probability of a correct detection, even after the image has been subjected to attacks.

2.1.2 Non-linear Quantization Strategy

The quantization schemes perform non-linear modifications and detect the embedded message by quantizing received samples.

2.1.2.1 Least Significant Bits (LSB) Method

Least significant bits (LSB) method is the most straight-forward method of quantization watermarking embedding scheme [24]. It operates in the spatial domain and replaces the least significant bits (LSB) of the cover image by quantized watermark bits. The watermark is embedded multiple times. Even if most of embedded watermarks are lost due to attacks, a single surviving watermark would be considered a success. LSB has proved to be a simple and fairly powerful tool for steganography, however, it lacks the basic robustness that watermarking applications require.

2.1.2.2 Quantization Index Modulation (QIM)

Quantization Index Modulation (QIM) refers to a class of data hiding schemes that exploit Costa's famous findings by embedding information in the choices of quantizers [25]. Over the past few years, QIM-based data hiding has received increasing attention from the data hiding community. This class of techniques embeds the watermark in a cover image through quantization; different quantization vectors are used to embed different watermark values. Recently proposed QIM schemes include Chen and Wornell's QIM and dither modulation [26], Eggers *et al.*'s scalar Costa schemes (SCS) [27], and application tailored implementations [28, 29].

Chen and Wornell argue that QIM structures are optimal when the watermark is energy-constrained and QIM methods can be considered to be better suited for data hiding applications than spread-spectrum-based watermarking methods [26].

2.2 Embedding Domain

A watermark can be embedded into a cover image in a spatial domain. Alternatively, a watermark embedding operation can also be carried out in transform domains, such as the discrete Fourier transform (DFT) domain, the full-image (global) discrete cosine transform (DCT) domain, the block-based DCT domain, the Fourier-Mellin transform domain, or the wavelet transform domain. Transform domain watermarking techniques apply some invertible transforms to the cover image before embedding watermark. Then, the transform domain coefficients are modified to embed the watermark and finally the inverse transform is applied to obtain the marked image. The major differences between the watermarking schemes in frequency domain lie in the different coefficient selection strategies.

2.2.1 Spatial Domain

The basic linear addition of spread spectrum signals can be applied to embedding one bit watermark in spatial domain with blind detection. Several proposed spatial domain systems are presented as following. The basic scheme is extended in some cases [30-32].

2.2.1.1 Patchwork

Other than LSB, Patchwork is another spatial domain technique designed to imperceptibly embed a single bit of information in a cover image. Patchwork embeds a watermark by changing the statistical distribution of luminance values in a set of pseudo-randomly selected pairs of image pixels. Patchwork is an elementary and nonrobust method [30].

2.2.1.2 Multiple-bit Additive Algorithms

Most watermarking applications require more than one bit of information to be embedded. The information rate of the watermarking system can be increased by introducing additional watermarks. This technique is known as direct message coding. The multi-bit message can be embedded into a cover image by adding watermarks representing individual bits of the multi-bit message to the cover, one by one [33]. Generally, watermarks representing individual bits of a multi-bit message are first combined together into a single watermark representing the whole message, and then added into the cover image.

Watermarks can be combined together in a couple of different ways. They could be tied together in such a way that any individual tile is a watermark representing individual message bit. This is equivalent to the space division multiplexing. For example, the watermark can be divided into blocks, and each block represents a message bit. Alternatively, frequency division multiplexing could be used where watermarks representing individual message bits would be placed into disjoint frequency bands. An approach analogous to Code Division Multiple Access (CDMA) in spread spectrum

communications could be applied in a watermarking application [15, 16]. In this approach each bit is spread across the whole image. The watermarks representing individual bits can be combined together without interfering with each other because they are selected to be mutually orthogonal [18].

2.2.2 DCT Domain

DCT domain watermarking can be classified into Global DCT watermarking and block based DCT watermarking.

2.2.2.1 Global DCT Watermarking

One of the first algorithms presented by Cox *et al.* [16] uses the global DCT approach to embed a robust watermark in the perceptually significant portion of the Human Visual System (HVS). Embedding in the perceptual portion of an image has its own advantages because most compression schemes remove the perceptually insignificant portion of the image. In the spatial domain it represents the least significant bits (LSB). However, in the frequency domain it represents the high frequency components.

2.2.2.2 Block DCT Watermarking

The main steps of any block based DCT algorithm can be simplified as following, as summarized by Potar *et.al* [11]:

1) Segment an image into non-overlapping small blocks, e.g. 8*8 block;
2) Apply forward DCT to each of these blocks;

3) Apply some block selection criteria (e.g. HVS);

4) Apply coefficient selection criteria (e.g. highest);

5) Embed watermark by modifying the selected coefficients;

6) Apply inverse DCT transform on each block.

The main difference among most algorithms is either in the block selection criteria or coefficient selection criteria.

2.2.2.3 Perceptual Modeling Strategy

Based on the perceptual modeling strategy incorporated by watermarking algorithms, watermarking algorithms in the DCT domain could be classified as algorithms with no perceptual modeling, implicit perceptual modeling or explicit perceptual modeling.

1) No perceptual modeling

Such algorithms do not incorporate any perceptual modeling strategy while embedding a watermark. Examples can be found at [34, 35].

2) Implicit Perceptual Modeling

These algorithms incorporate the transform domain properties for perceptual modeling [36, 37]. Those high-frequency coefficients in transformation domain are selected, because they allow strong watermarks to be embedded and result in least perceptual distortion [38]. DC components satisfy this criterion and hence they can be used. They also select those coefficients which are least changed by common image processing attacks like low-pass filtering and noise addition. Low frequency AC components as well as high magnitude DC components satisfy the above criteria and can be selected.

3) Explicit Perceptual Modeling

The HVS properties for perceptual modeling are incorporated in such algorithms [38-40]. HVS models allow us to increase or decrease the strength of a watermark because it takes into account the local image properties, for example, contrast, brightness variance, etc.

2.2.2.4 Comparison of DCT with Spatial Domain

DCT based watermarking techniques are more robust compared to spatial domain watermarking techniques [16]. Such algorithms are robust against simple image processing operations like low pass filtering, brightness and contrast adjustment, blurring etc. However, they are difficult to implement and are computationally more expensive. At the same time they are weak against geometric attacks like rotation, scaling, cropping, etc.

2.2.3 DWT Domain

Discrete Wavelet Transform (DWT) based watermarking schemes follow the same guidelines as DCT based schemes. However, the process to transform an image into its transform domain varies and hence the resulting coefficients are different. Wavelet filters are used in wavelet transforms to transform images. Among the most commonly used filters are Haar Wavelet Filter, Daubechies Orthogonal Filters and Daubechies Bi-Orthogonal Filters [41].

As mentioned earlier, blind detection does not require the original image for detecting the watermarks, however, non-blind detection requires the original image.

Existing wavelet based watermarking algorithms can be classified as blind detection [41-43] and non-blind detection [44-46].

2.2.3.1 Advantages and Disadvantages of DWT over DCT

DWT and DCT are the most popular domains for watermarks. In general, DWT produces images with more invisible watermarks and higher storage capacity [47]. DWT is superior to DCT in terms of the applicability in the HVS [11]. With the standardization of JPEG-2000 and the decision to use wavelet-based image compression instead of DCT-based compression, watermarking techniques operating in the wavelet transform domain have become more attractive to the watermarking research community. Moreover, analysis is provided to show that for common attacks such as spatial cropping and compression, the wavelet-domain, which tends to isolate these distortions, is one of the best in which domains to embed the information.

However, the computational complexity of DWT is higher than DCT [42]. As Feig [48] pointed out it only takes 54 multiplications to compute DCT for a block of 8*8, unlike the wavelet calculation which depends upon the length of the filter used, and is at least one multiplication per coefficient.

2.2.4 DFT Domain

Adding a watermark to the DFT magnitude coefficients was proposed by O'Ruanaidh *et al.* [49]. DFT domain has been explored by researchers because it offers robustness against geometric attacks like rotation, scaling, cropping, translation etc.

The DFT of a real image is generally a complex value, which results in the phase and magnitude representation of images. The best location to embed the watermark in the DFT domain is the mid-frequency [50]. The major advantage of DFT over DWT and DCT is that DFT is rotation, scaling and translation (RST) invariant. Hence DFT can be applied in watermarking schemes against geometric distortions, whereas in DCT and DWT and in the spatial domain, watermarks are difficult to extract after geometric distortions.

2.3 Attack Analysis Methods

Research in digital watermarking has progressed along two paths. While new watermarking technologies are being developed, some researchers are also investigating different ways of attacking digital watermarks. Common attacks to watermark usually aim to destroy the embedded watermark or to impair its detection. Since the more specific information known about the family of possible attacks, the better a system can be designed to resist it, some researchers have selected to work on modeling and resisting attacks on watermarks. These approaches concentrate on qualifying and quantifying attacks and their effects, and involve developing countermeasures against them.

2.3.1 Blind Removal Attacks

Blind removal attacks aim at completely removing a watermark from a cover image regardless of secret keys. These approaches consider the inserted watermark as noise with

a given statistic and attempt to estimate the original cover image. Blind removal attacks as a means of robustness assessment are widely investigated in traditional watermarking studies. These include addition of noise, compression/filtering attacks, geometric distortions, etc. Early researches explored the counter-measures against attacks such as compression, filtering or noise addition. Selection of appropriate coefficients for watermark embedding in a transformation domain is one of the well established counter-measures. More recently, geometric distortions have been of great interest to watermarking specialists. Proposed counter-measures include embedding a template, i.e. an extra signal for synchronization of the embedder and detector [51], and embedding a watermark signal in an invariant domain [49], as well as synchronizing by image self-registration [52, 53].

2.3.2 Attacks Based on Key Mapping Function Estimation

This class of attacks can be considered as security attacks. Recently, some attempts address the concept of watermarking schemes from a cryptanalytic point of view. All the information about watermarking schemes is public, and security relies only on the use of secret keys. If the secret key is known by attackers, the watermarking scheme's security is broken. For attackers, it may not be possible to discover the secret key, since usually the mapping function (input is the secret key, and the output is the watermarked image) designed so as to not be easily invertible. However, knowledge of the mapping function may be enough for the attacker's purpose. For example, when the attacker has knowledge

about the watermarking scheme used, he can try to obtain an estimate of the transformation through observation of the outputs of the embedder and/or decoder.

2.3.2.1 Averaging Attacks

The main flaw of spread-spectrum schemes in terms of security is that the same pseudo-random pattern is embedded repeatedly. Statistical averaging attacks are based on the fact that, if multiple images with the same embedded watermark are available, it is possible to estimate the watermark by averaging all those images.

As a solution, several watermarks are randomly selected to prevent averaging attacks [54]. Another possible solution [55] recognizes the security advantages of using image-dependent keys. The authors present a method for generating a Gaussian vector, which is depending on both a secret key and a robust hash function of the cover image [55].

2.3.2.2 Watermark Removal Filtering

Knowing the algorithms, attackers can resort to more powerful attacks, since they are able to play in the embedding domain. More sophisticated attacks than classical content transformations rely on noise removal filtering, in particular if suitable statistical models of original features and watermark images are available. For example, Voloshynovskiy *et al.* [56] have developed a watermark removal filter based on maximum likelihood or maximum a posteriori probability criteria. In practice, attackers look for the best approximation of the original document, by assuming that the watermark can be viewed

as disturbing noise. Similarly, Wiener filtering can be adopted to try to separate the watermark and the host document.

A possible countermeasure, suggested by Su *et al.* [57], is to follow the Power Spectrum Condition stating that the power spectrum density of the watermark should be shaped like the one of cover images. Another possibility, proposed by Pateux *et al.* [58], is to embed the watermark signal and then to self-attack the resulting watermarked image by a Wiener filter. This highly diminishes the efficiency of watermark removal filters.

2.3.2.3 Sensitivity Analysis Attacks

Sensitivity analysis attacks constitute a powerful family of watermark removal attacks. They exploit the vulnerability in some watermarking protocols: the attacker's unlimited access to the watermark detector. For example, attackers may be motivated to remove watermarks from a watermarked home video copy in order to produce an unlimited number of illegal copies and resell them. Attackers make use of the detector to extract information about the watermark and subsequently "remove" it. Since most detectors decide on the presence or absence of the watermark by comparing the correlation value with a threshold, attackers decide to find the threshold using test images differing from each other by changes in luminance of a few pixels. With this knowledge of the detector, the image space can be divided into those areas that give correlator outputs less than the threshold and those are greater than the threshold. So attackers can find the tangent to the curve that divides these two regions, which involves testing all pixel positions. Now that the tangent is known, attackers can subtract just enough out of the luminance of the original image to give a negative detection result and very small perceptual difference. This process can be iterated if the attacker is not satisfied with the perceptual damage.

Several approaches to resisting the sensitivity attack have been proposed. The countermeasure [59] is based on randomization of detected results in a defined interval between "watermarked" and "not watermarked" regions [60]. It also suggests converting the decision boundary to a fractal curve [61]. However, the fractalization does not change the outline of the decision boundary; therefore, it is still possible to estimate the embedded watermark signal in order to destroy it.

Chapter 3

Counter-attacks of Geometric Distortion

3.1 Introduction

A watermark's resistance to geometric attacks is the ability to withstand an arbitrary displacement of all or some of its pixels by a random amount. It is a fundamental issue in watermark system design [62]. A geometric attack can render virtually any watermarking application useless. The robustness of watermarks to geometric manipulations can be compared to losing synchronization in a communication system, since the detection of marks requires a synchronization step to locate the embedded mark in the content. Unintentional geometric attacks include image-processing manipulations such as scaling

images for a web site, printing and scanning marked documents, changing a digital video's aspect ratio, and cropping an image to extract a region of interest. Resizing and rotating are also basic manipulations in image edition and require a synchronization step for the detection of marks. Although specific schemes try to circumvent these attacks, a significant portion of existing algorithms fail to survive even such apparently simple modifications.

Most of the proposed techniques in the early 1990's can be classified, based on the frequency domains in which watermarks are embedded, into methods using DCT domain, DWT domain, and DFT domain, etc. Frequency domain based watermarking algorithms have been extensively studied to improve the robustness against certain signal processings, such as JPEG and/or other compression techniques, noise addition, and lowpass filtering [63, 64]. However, reliable detection of frequency-based watermarks is impeded when synchronization is lost as a result of geometric transformations. Although watermarking using spread spectrum in a transformed domain is very resistant to amplitude distortions and additive noise, it becomes fragile if the starting point for decoding is flost. Counter attack methods against geometric distortion have attracted more attention from recent watermarking research groups.

Compared to methods using frequency domains, spatial watermarking methods are a better option for prevention of geometric attacks because they target at specific locations in images. Although spatial watermarking is less resilient to certain signal processing attacks, it allows extraction of information related to the spatial distortions of pixels. With this information, the image can be resynchronized after undergoing geometric transformation.

In the past ten years, the understanding of geometric attacks has been significant improved. This chapter attempts to review existing watermarking methods against geometric transformations. More literature reviews about geometric attacks can be found at [62, 65].

3.2 Effects of Geometric Distortions

Effects of geometric distortions on watermarking embedding schemes are shown in Figure 3.2.1. The image rotation leads to loss of synchronization, i.e. the detector fails to locate the watermark.

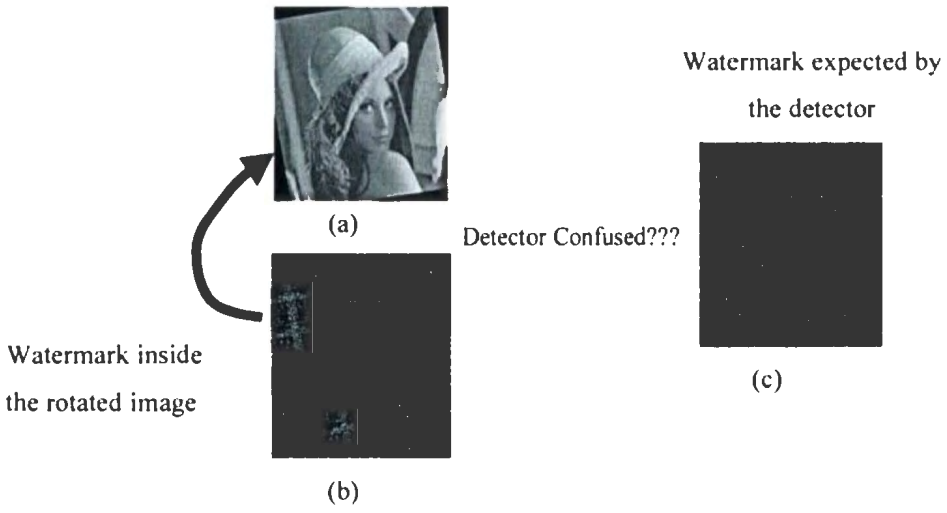


Figure 3.2.1: Global geometric distortion effect

In Figure 3.2.1, (a) represents the watermarked image, (b) shows the inserted watermark and (c) is the watermark expected by the detector. The detector lost the synchronization after the geometric distortion. The correlation cannot be performed

because the randomly generated sequence and the embedded sequence are not synchronized as shown in (b) and (c).

The additive and quantization schemes as major watermarking embedding schemes are affected in different manners, as illustrated below. A majority of proposed additive watermarking algorithms operates on principle analogous to spread-spectrum communications. A pseudo-random sequence, which is generated using a secret key, is inserted into images. During extraction, the same pseudo-random sequence is correlated with the estimated pattern extracted from images. The watermark is said to be present if the computed correlation exceeds a chosen threshold value. Among this general class of watermarking schemes, there are several variants that include choice of a specific domain for watermark insertion, e.g. spatial, DCT, wavelet, etc; and enhancements of the basic scheme to improve robustness and reduce visible artifacts. The computed correlation depends on the alignment of the pattern regenerated and the one extracted from the image. Thus proper synchronization of the two patterns is critical for the watermark detection process. Typically, this synchronization is provided by the inherent geometry of the image, where pseudo-random sequences are assumed to be placed on the same image geometry. When a geometric manipulation is applied to the watermarked image, the underlying geometry is distorted, which often results in the de-synchronization and failure of the watermark detection process. Similarly, in a quantization scheme, the marked components cannot be located due the fact that initial locations depend on external coordinates. Thus the decoding of the mark is impossible.

3.2.1 Classification of Geometric Distortions

Geometric manipulation ranges from simple scaling, rotation and aspect ratio changes to more complicated random geometric distortions. Generally, geometric distortions can be classified as global geometric distortion and local geometric distortion.

Global geometric distortions commonly appear in image manipulations, such as rotation, translation, cropping, and composition. These transformations are applied on the whole image, and in many ways can be easily represented by a mathematical operation.

Local geometric distortions are especially designed to desynchronize the mark without visual changes. Khun and Petitcolas [65] have developed a benchmark called StirMark containing different attacks. One of the first attacks developed by this program is composed of local random geometric distortions that permit us to defeat many classical watermarking schemes without visible alterations.

According to Kutter [66], digital watermarking embedding schemes can be classified into two generations. In the first generation the location of the watermark to be embedded is decided by pixels or coefficients of frequency transform domains, while in the second generation the location is determined based on the notion of image feature. In the following sections, major watermarking schemes against geometric distortions are presented, which are classified as counter-attacks of the first generation and the second generation.

3.3 Methods of the First Generation

The first generation watermarking methods are characterized by image geometry searching for recovery of synchronization information. Four major methods in this category are discussed below.

3.3.1 Exhaustive Random Search

A geometric transformation such as rotation or scaling can be modeled as an affine transformation function with several parameters. Given enough sets of matching points, the function parameters can be derived. One obvious candidate solution to the synchronization problem is to perform an exhaustive random search over the space containing the set of acceptable attack parameters. By doing so, the solution to the synchronization problem will be the one that optimizes a certain cost function, such as the root-mean-square (RMS) error between the original and watermarked image or the RMS error between coordinate values for corresponding pixel locations in the original and possibly attacked image. The search space cardinality determines the resolution achieved by the synchronizer and also the computational cost of the performing. The larger the search space, the more accurate the synchronizer outcome, but it also requires more computation [62].

3.3.2 Template-based Synchronization

Another solution to counter geometric attacks is to identify the attack's transformation by retrieving artificially embedded templates. Templates are reference patterns known at both the embedder and detector, which are added to the image in addition to the watermark. The detector can synchronize the watermark by using knowledge about the template to identify geometric transformations.

Template-based synchronization has been proven to be effective under global attacks. The template can be inserted into both spatial and frequency domain. Templates embedded in the Fourier transform domain can be applied to render the method robust against general linear transformations, as described by Pereia and Pun [67]. In their method, templates are localized in a ring-shape area corresponding to the middle frequencies of the image spectrum. Eight evenly distributed coefficients are selected and their magnitudes are increased to generate templates. At the detection stage, the location of the template can be detected by searching eight maxima value within the ring area. The affine transformation can be identified by matching the initial location of templates and the detected location of local maxima. Fleet and Heger [68] use watermark projection on a set of sinusoids that appear as peaks in the frequency domain and these dots are used to synchronize the watermark.

However, because these peaks are easy to see, an attacker can identify them and then easily rip the template off the watermarked image. Another problem with this solution is that, because it requires the insertion of a template in addition to the data-carrying watermark, this approach is likely to reduce the image fidelity. Furthermore, all watermarked images share a common template and therefore are susceptible to collusion

attacks, which estimate and remove the templates from the watermarked images and thus restrict the invertibility of any geometric distortions [69].

3.3.3 Periodic Insertion of the Mark

An interesting approach to counter geometric distortion is to add redundancy during the embedding process. This redundancy can be used to localize the position of the signature and to improve the detection. This approach doesn't use templates but relies on the watermark's autocorrelation properties to achieve synchronization. Generally speaking, the watermark is designed such that its autocorrelation function contains several peaks. On the receiver-end side, the decoder correlates the received watermarked image with itself and uses the knowledge about the autocorrelation function's periodic nature to synchronize the watermark.

Kutter [70] uses space diversity (that is, vertical and horizontal shifts to embed repeated versions of the watermark) to estimate the attack parameters and invert them before detection. A periodic mark is embedded in the luminance of images. A cross-correlation function of the image allows the localization of the different peaks generated by the periodic mark, and consequently the identification of the geometric transform. Hartung *et al.* [71] also devises a method that periodically inserts the watermark to circumvent Stirmark attacks.

This method has great potential, but it is not flawless. Watermark detection in this method is dependent on the successful identification of the geometric distortion and the detection of the watermark after inversion of the distortion, and thus involving additional process and reducing the invertibility of geometric distortion [72].

3.3.4 Normalization Methods

Yet another approach to address geometric distortions is the "normalization" of a cover image prior to watermark embedding. After embedding, the image is restored to its original geometric state prior to distribution. Upon receipt, the image is again normalized prior to detection. Unlike scaling to a canonical size, the normalization must be invariant to the expected geometric distortions.

Images are normalized by their geometric moments [73]. In this way, watermarks are hidden by modifying image content iteratively to produce the mean value of several invariant moments in a predefined range. The watermark detector verifies the presence of the watermark by checking the mean value of these moments. This scheme can resist orthogonal transformations and general affine transformations.

However, since the image normalization process is applied to the entire image, it would be sensitive to cropping and local region distortion [52].

3.3.5 Embedding Domain Invariance

A more elegant approach to achieve robustness against loss of synchronization is to use transformations that map the image information into an invariant domain. The most commonly used transformations depend on the properties of Fourier transformation. Combining a Fourier transform with a log-polar map results in invertible rotation, translation, and scale invariant representation [74] . However, the exponential nature of the inverse log-polar mapping causes a loss of image information in the discrete space [49] .

Some other domains are invariant to translation alone. It's well known that DFT domain's magnitudes are invariant against translations in the space domain. Lin *et al.* [72] have presented a method that achieves robustness to global RST attacks. The method exploits the translation invariance of the DFT magnitude spectrum and the properties of log-polar mapping. Their method embeds the watermark in locations along the log-radius axis obtained by mapping the DFT magnitude spectrum to a log-polar coordinate system. Thus, rotations are mapped to the watermark's cyclic shift. Since the detector is based on the normalized correlation coefficient, it compensates for scaling attacks. However, this DFT based method lacks robustness to cropping and localized attacks.

3.4 Methods of the Second Generation – Feature-Based Methods

The second generation method is also called "content-based watermarking scheme". Media contents represent an invariant reference for geometric transformations so that referring contents can solve the problem of synchronization. The location of the signature is not related to image coordinates, but image semantics. This method does not rely on the presence of a template, which an attacker can erase to confuse the watermark decoder, but rather on salient image features. The synchronization based on image features (edges, corners, connected components, texture, and so on) relies on the ability to identify certain feature points in an image before and after an attack. If enough points establish correspondence, it is possible to invert the attack.

In the feature point based approach, the feature points detected in the original image are used to form local regions for embedding. At the detection end, the feature points are expected to be robustly distributed at the corresponding positions. The common framework is that some kind of image units such as blocks [75], meshes [52, 53], or disks [76] are extracted as carriers for embedding. The construction of these patches is based on the extracted feature points. Each image unit / patch in an image can be treated as a small image and the watermark is embedded into each image patch.

Patch locations are extracted by clustering feature points using the adaptive K-mean clustering method and retrieving several large regions where most feature points are located [77]. These regions are fit by ellipsoids and their bounding rectangles are used as the patch to embed or detect the signature. However, segmentation based feature selection is sensitive to some image modifications, such as image cropping and translation of the image.

Alghoniemy and Tewfik [9] use edges as feature points to estimate attack parameters. They estimate edges from the image wavelet maxima computed via a multiresolution-level decomposition of the image. Based on the computation of the average edge standard deviation ratio and the average edge angle difference, they estimate the scaling factor and rotation angle, respectively. This method doesn't rely on the original cover image's presence during detection. However, this method requires prior information about the cover image.

In Bas *et al.*'s approach [53], patch locations are based on salient feature points and patches are constructed by feature points' Delaunay tessellation. In order to formulate patches for watermark embedding and detection, feature points are extracted by applying a Harris corner detector. The set of extracted feature points is decomposed into

a set of disjoint triangles through Delaunay tessellation. If the set of extracted feature points in the original image and distorted images is identical, applying Delaunay tessellation will be an efficient method to divide the image. Delaunay tessellations are invariant to spatial filtering and geometric distortions, in particular scaling and rotation. Each triangle can be treated as a patch. The signature is embedded into the patch by applying a classical additive watermarking method on the spatial domain.

Tang and Hang [52] has introduced a synchronization approach by using the intensity-based feature extractor and image normalization. In general, the objects in the normalized image are invariant to small image modifications and this approach focuses on this fact. They use a method called the Mexican hat wavelet scale interaction, which determines feature points by identifying the intensity change of the image and is more robust to spatial distortions. Subsequently, disks are constructed with the extracted feature points being their centers and normalized in order to be invariant to rotation, translation, and partial filtering of the image. They use these normalized disks as patches for watermark embedding and detection.

Rongen *et al.* [78] has presented a method drawing several lines crossing the images. Locations of those lines depend on whether large percentage feature pixels are near a line. Finally, it applies small modifications so most feature points lie on the lines. The watermark can be detected by checking whether most salient points can be found on specified lines.

In general, methods based on image features are robust to errors and attacks. Since the framework depends on the repeatability and accuracy of the feature point detectors, most of the algorithms fail to ensure repeatability under a broad range of image

processing operations. Moreover, these methods are difficult analyze theoretically. Hence, their use in commercial applications might be restricted [79].

Chapter 4

Data Hiding Scheme with Geometric Distortion Estimation

4.1 System Introduction

The motivation for the proposed scheme is derived from a specific requirement for robustness against geometric distortion in data hiding applications. A general overview of the scheme as well as details of involved techniques is given below.

4.1.1 Scheme Motivation

In order to resist to geometric attacks, it is necessary for a recovering algorithm to determine what operation (translation, rotation) has been applied to produce tampered images. Image registration can be an efficient recovering algorithm, which is mapping the received image to an original cover image to determine locations where the watermark is embedded. Image registration is a minor problem if the original image is available to the watermark detector. However, in watermarking applications, it is usually impossible to retrieval the original cover image among the huge image database, since detectors do not have access to the original cover image. So the registration process can not be performed. In this case, some authors have proposed different methods to estimate transformations that the image has undergone and to reverse their effect. To estimate these transformations, some reference is needed. Invariant image contents (feature points, edges, etc.) can fulfill this requirement. Invariant image contents in cover images can be used as *reference information*. The locations in distorted images can be determined as *actual information*. Distortion transformations can be estimated without accessing the cover image, if both reference information and actual information are available.

These methods of estimating distortion transformations by reference information and actual information appear advantageous. However, there are some difficulties when these methods come to practical applications. Although these methods do not require the original cover image, they still need some prior information, reference information, about the cover images. In other words, geometric distortion estimation is not feasible technologically, if reference information is absent.

This work aims to investigate the possibility of embedding the reference information as part of the watermark to cover images. The reference information contained in the watermarked image can be easily extracted and decoded by the detector. Therefore, the geometric distortion, scaling and rotation, can be achieved by comparing the reference location and actual location without accessing any information about the cover images.

4.1.2 Presentation of the Proposed Algorithm

As is well known, RGB images have three color spaces, red, green and blue. Each color space can be considered as one independent channel [10]. Two channels are selected in this work. One channel (communication channel) carries a large amount of information to be embedded, and the other one (synchronization channel) is used to check whether the image has been accidentally rotated or rescaled.

The scaling factor and rotation angle can be estimated from feature points. The scaling factor can be approximated by comparing the deviation of the feature points from mean of feature points before and after scaling. Rotation angle can be calculated by comparing the average angles between feature points in the first quadrant before and after rotation. The deviation and average angles of original feature points can be selected as reference information.

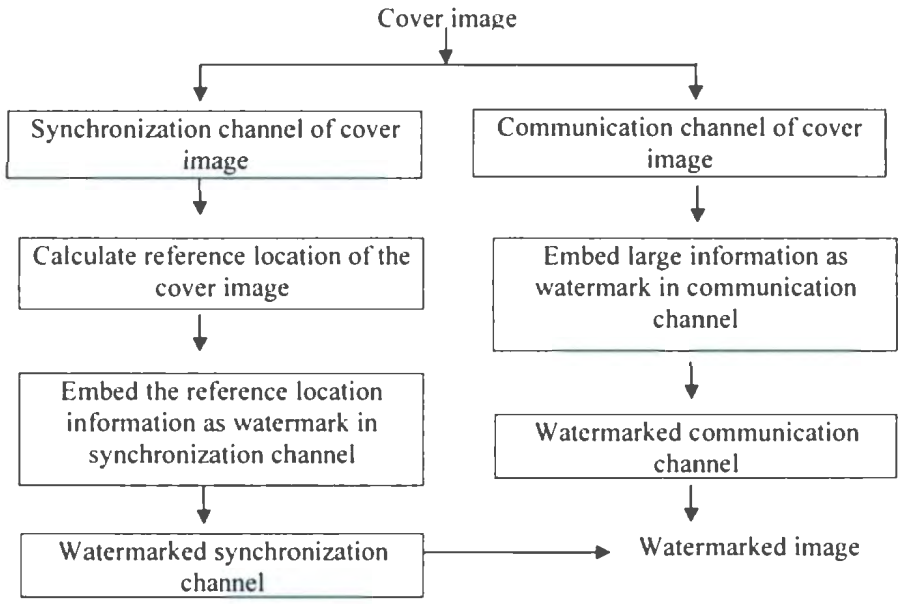


Figure 4.1.1: Embedding stage

The main flow charts for the embedding and extraction stages are shown in Figure 4.1.1 and Figure 4.1.2, respectively. In the embedding stage, reference information, which is based on statistics information related to feature points, is embedded as a watermark into the synchronization channel. Content-based watermarking scheme, which has very high robustness against geometric distortion, is used in the embedding and detection of the synchronization channel. While many data hiding schemes are available for embedding the information to be hidden into the communication channel, this work focuses on design and evaluation of the embedding scheme for a synchronization channel.

In the detection stage, the geometric distortion is estimated in the synchronization channel before the extraction of embedded information in the communication channel. The synchronization information of the transformed image (actual information) can be calculated and that of original image (reference information) can be extracted. With the

availability of synchronization information for both original and distorted images, the distortions can be easily estimated and then corrected.

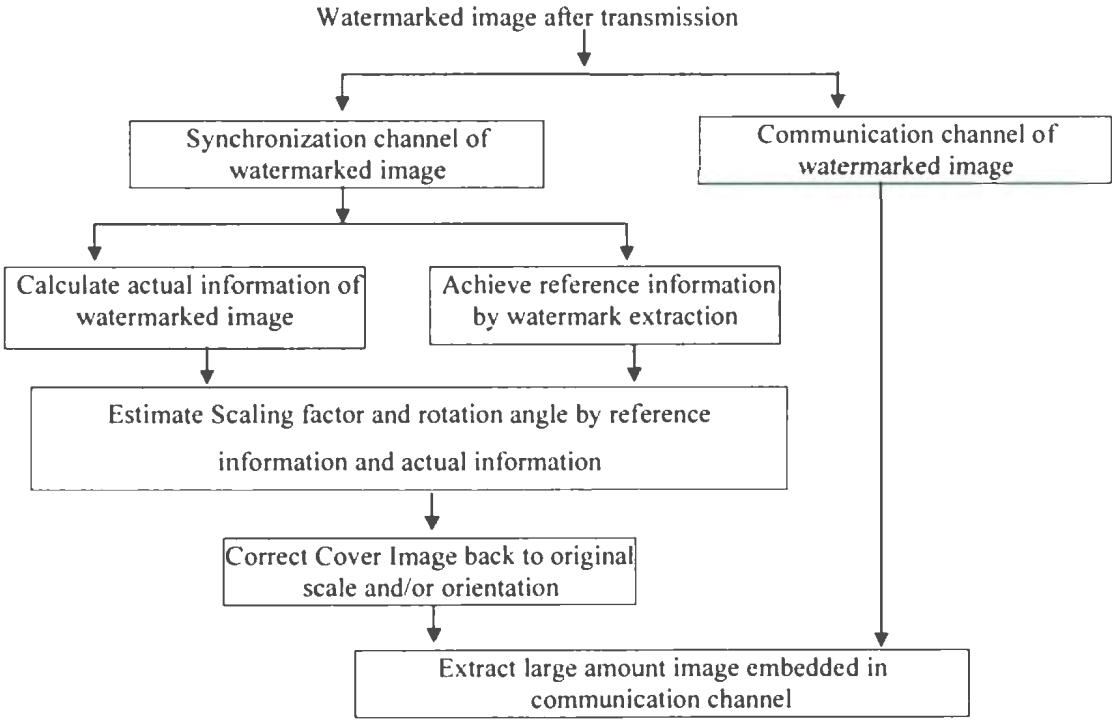


Figure 4.1.2: Detection stage

Due to the robustness requirement of the reference information, a content-based watermarking scheme is applied at synchronization channel as shown on Figure 4.1.3. Content-based synchronization markers are essentially composed of three building blocks. First, a set of feature points are extracted. Then, elementary patches, triangles, are formed based the set of extracted feature points. Finally, the watermark, containing the reference information, will be inserted into each triangle repeatedly. In order to fit within different shapes of individual triangles, the reference location information is spread into a pre-defined standard triangle, e.g. an isosceles right triangle, and the standard triangle is warped into the shape of individual triangles.

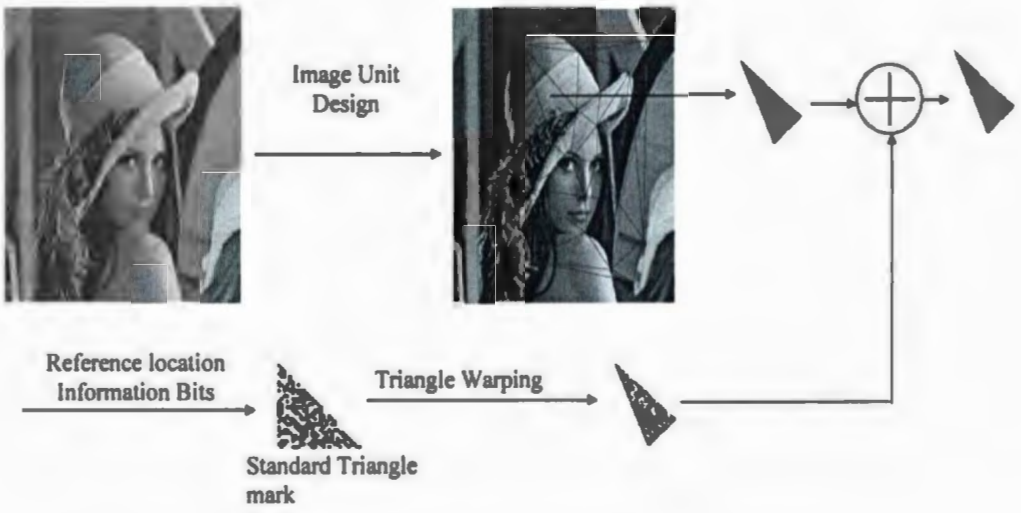


Figure 4.1.3: Synchronization channel embedding scheme

Several vital techniques of the system, such as reference information generation, robust feature point extraction, and element patches (triangles) formation are presented in next sections. The whole embedding and extraction diagram is described as well.

4.2 Reference Information

Image rotation and image rescaling are the most common geometric distortions during image manipulation. Masoud [9] presented a method to estimate the scaling factor of a previously scaled image and the angle by which the image has been rotated. The main idea can be summarized as follows.

The average distance of feature point (i.e., (x_i, y_i), as shown in Figure 4.2.1) from the center of gravity of the extracted feature points, i.e., (x_o, y_o) is:

$$\sigma = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(x_i - x_o)^2 + (y_i - y_o)^2},$$

where N is the total number of feature points

The scale factor γ can be estimated by:

$$\gamma = \sigma_s / \sigma_o .$$

Where σ_s and σ_o represent the average distances of feature points in the distorted image and in the original image, respectively. The average distance in the original image (σ_o) is selected as reference information for the scaling factor.

The average of angles θ_i'' (as show on Figure 4.2.2) which are the angles in the first quadrant of the feature points in the original image is:

$$\theta'' = \frac{1}{N}\sum_{i=1}^{N}\theta_i'' .$$

The rotation angle estimation θ is given by

$$\theta = \theta'' - \theta',$$

where θ'' and θ' respectively represent the average angles in the first quadrant of the feature points of the original and the distorted image. The average angle of the original image (θ'') can be selected as reference information for the rotation factor.

Figure 4.2.1: Average distance of feature points from the gravity center

Figure 4.2.2:Rotation angle estimation

In the original work of estimating the rotation angle and scaling factor [9], Masoud assumes that the decoder has prior information regarding the original image

information, reference information σ_u and θ''. That is to say that both the encoder and the decoder should agree beforehand upon specific σ_u and θ'' values for which the watermark is inserted and detected at these values. However, it is not practical since the decoder usually does not have prior information about the original image. This thesis presents a solution in which reference information σ_u and θ'' can be embedded as part of watermark. Most of the effort of this thesis focuses on how to embed and extract the reference information.

4.3 Feature Point Extraction

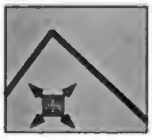
Feature points extraction plays a key point in this scheme. The robustness of both the reference information and content-based watermarking scheme is based on robust feature point extraction. To improve the robustness of feature point extraction, the Harris-Laplacian feature point detector is modified.

4.3.1 Harris Corner Detector

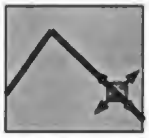
Feature point extraction has been well studied by some image processing research groups. Feature point extraction was first developed for computer vision and reconstruction, but is also employed in data-base retrieval as a descriptor of images. Edge and corner detectors are popular in image processing and can be employed for feature point extraction. Major feature point extraction techniques used in watermarking applications

such as the Harris corner detector, the Susan corner detector, and the Achard-Rouquet detector, are compared in [53]. The Harris corner detector has shown the best geometric stability under different transformations, such as image rotation, illumination transformation and perspective deformations [53, 80].

In general, the Harris corner detector generates the second moment matrix using image gradients and then combines eigenvalues of the moment matrix to compute a corner-strength, whose local maxima indicate corner positions. The second moment matrix is $E_{x,y} = (x,y)\mu(x,y)^T$ with $\mu = \begin{bmatrix} L_{x,x} & L_{x,y} \\ L_{x,y} & L_{y,y} \end{bmatrix}$, L represents image gradient alone the x and y axis. $E_{x,y}$ can be considered as a local auto-correlation function of the image with a shape factor μ as depicted in Figure 4.3.1. Corners can be distinguished if there are significant changes in all directions. Harris [81] gave a new definition of the detector function based on eigenvalues λ_1 and λ_2 of the second moment matrix $E_{x,y}$. To avoid computing the eigenvalue of μ, the new criterion is based on the trace and determinant of μ : $Tr(\mu) = \lambda_1 + \lambda_2 = L_{xx} + L_{yy}$, $Det(\mu) = \lambda_1 \cdot \lambda_2 = L_{xx}L_{yy} - L_{xy}^2$. The corner-strength can be represented by $R_H : R_H = Det(\mu) - kTr^2(\mu)$ where parameter k is an empirical constant normally in the range of 0.04-0.06 [81]. Feature point extraction is achieved by applying a threshold on R_H and searching for local maxima.



(a)Flat Region: No change in all directions

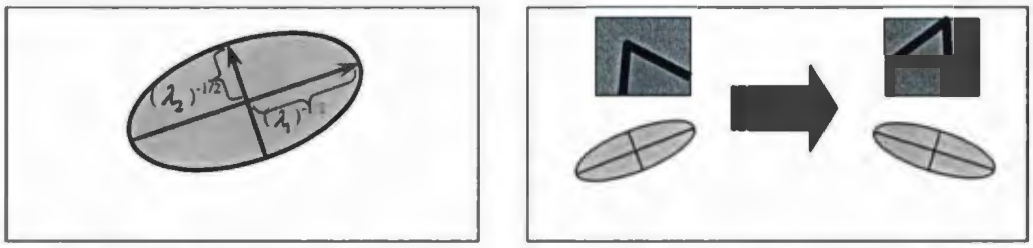


(b)Edge: No change along the edge

(c)Corner: Significant change in all directions

Figure 4.3.1: Auto-correlation function in different cases

The second moment matrix $E_{x,y}$ is represented by an ellipse, which is formed based on eignvalues λ_1 and λ_2 of $E_{x,y}$ as depicted in Figure 4.3.2(a). After the image rotation, the orientation of this ellipse is changed but the shape remains the same, since the eigenvalue of $E_{x,y}$ are invariant with respect to image rotation [80, 82]. Hence, corner detector is invariant with respect to image rotation.



(a) Ellipse $E_{x,y}$

(b) Ellipse rotates but its shape (i.e. eigenvalues) remains the same

Figure 4.3.2: Harris corner detector is invariant to image rotation

However, the Harris corner detection is sensitive to changes in image scaling as illustrated in Figure 4.3.3. The line in Figure 4.3.3 (b) represents the one from Figure 4.3.3 (a), but in the presence of a scale factor of 0.5. All points along the line in (a) will be classified as edges. In (b), after scaling, edges of the line are getting sharper and one of them turns to be a corner.

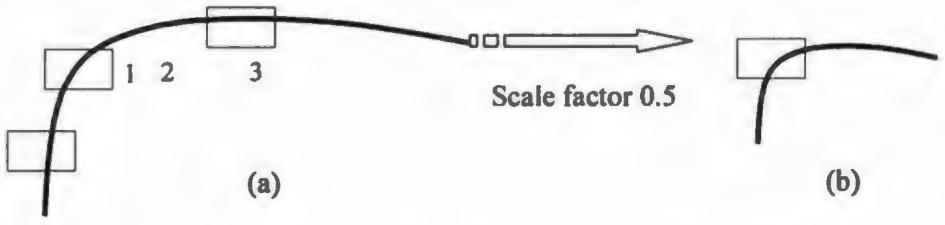


Figure 4.3.3: Harris corner detector is non-invariant to image scale

The feature points can be lost significantly when an image undergoes geometric distortions if the Harris corner detector is directly applied on the image. Therefore, some measurements must be employed to improve the robustness of the feature point detector.

4.3.2 Scale Space Theory Improvement

Several authors have proposed some ways to improve the feature point detector when using the Harris corner detector in watermarking applications. A simple enhancement is done by using a pre-filtering smoothing operation [53]. An $n \times n$ mask represented by $M_n = \frac{1}{n \times n}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & 1 \\ 1 & \cdots & 1 \end{bmatrix}$ improves the robustness against signal processing noise. However, this average filter may introduce ringing problems into the image as it corresponds to a sinc function in the frequency domain, which allows certain higher frequencies to pass through the filter. Another way has emerged which performs Harris corner detector on both a cover image and a transformed image [10]. A set of feature points existing in both images are selected as feature points. The robustness is improved by obtaining stable points in both the cover image and the transformed image.

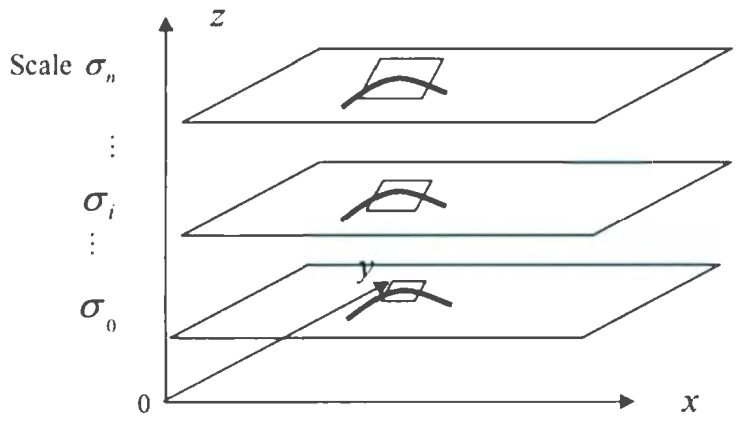


Figure 4.3.4: Scale space representation of Harris-Laplace

In order to improve the robustness of the feature point extractor, the scale space theory is adopted in this scheme. In image indexing, retrieval, recognition applications, feature point extractor with scale space improvement has been studied to build robust feature descriptors. The Harris-Laplacian detector was first presented by Mikolajczyk [83] for object recognition. The descriptor detection is based on two steps.

The first step is to compute interest points by the Harris corner detector at several scale levels. The scale-space is computed using a Gaussian function, where an image is filtered by Gaussian functions of different scales and then different images are formed. A scale-space representation for the Harris function can be built as shown in Figure 4.3.4. Images at different scales are built along the z axis.

The second step is to select points which a local measure (the laplacian) is maximum along the z axis. The scale of selected feature points is called the characteristic scale. Local descriptors (such as ellipse, which are constructed based on the eigenvalue of the Harris corner detector) could be constructed at characteristic scales. Those descriptors can be applied as robust indices for image matching.

The Harris-Laplace detector has been proven to be invariant to image rotation, scaling, translation. However, the approach is an exhaustive search over feature points, if every feature point needs a search for the characteristic scale along the scale space. The Harris-Laplace detector is designed for object recognition, and the characteristic local structure is required. In this work, the Harris-Laplace detector can be simplified to improve the efficiency. The Harris-Laplace is used to keep corners which can be detected in the same location but on different scales.

First, the second moment matrix of Harris corner detector in scale space can be defined by:

$$\mu(x, y, \sigma_I, \sigma_D) = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} = \sigma_D^2 g(\sigma_I) * \begin{bmatrix} L_x^2(x, y, \sigma_D) & L_x L_y(x, y, \sigma_D) \\ L_x L_y(x, y, \sigma_D) & L_y^2(x, y, \sigma_D) \end{bmatrix}$$

where σ_I is the integration scale, σ_D is the differentiation scale

L_x, L_y are the derivatives computed in the direction x, y, respectively.

$$L_x(x, y, \sigma) = (\frac{\partial}{\partial x} g(\sigma)) * I(x, y), L_y(x, y, \sigma) = (\frac{\partial}{\partial y} g(\sigma)) * I(x, y)$$

The Gaussian kernel used here is a uniform scale-space kernel,

$g(\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$.

The matrix describes the gradient distribution in a local neighborhood of a point. The differentiation scale σ_D determines the size of the Gaussian kernels, which are used for computation of local derivatives. Averaged derivatives are then obtained by smoothing with a Gaussian window, the size of which is dependent on the integration scale σ_I.

In this work, the pre-selected scales $\sigma_i = \xi^i \sigma_0$, where $\xi^i, (i = 1 \cdots N, N$ as the maximum scale) is the scale factor between successive levels. The matrix $\mu(x, y, \sigma_i)$ is computed with the integration scale $\sigma_I = \sigma_i$ and the local differentiation scale $\sigma_D = s\sigma_i$, where s is a constant factor. The corner-strength R_H can be calculated by the trace and the determinant of this second moment matrix:

$$R_H = \det(\mu(x, y, \sigma_I, \sigma_D)) - \alpha trace^2 \mu(x, y, \sigma_I, \sigma_D)) .$$

Local maxima of R_H in the 8-neighborhood of a point determine the location of corner points. A threshold is used to reject those maxima where R_H is small.

Next, a more compact representation is achieved by selecting feature points existing at the same location at all scales. These feature points are scale-invariant.

Following is an example of detecting a feature point along different scales, where pre-selected σ_i =0.25, the number of total scales n is 4, the scale factor between successive levels ξ is 0.25 and the constant factor s =1. The integration scale $\sigma_I = \sigma_i$, and the differentiation scale $\sigma_D = s\sigma_i$.

Feature points extracted at different scales are presented in Figure 4.3.5(a). The four markers with different style and color represent the feature points extracted at 4 different scales. Only those feature points existing at all scales are selected, as highlighted red in Figure 4.3.5(b). Feature points are detected by checking whether two points correspond using the criterion that the error in relative location does not exceed 1 pixel in the coarse resolution image.

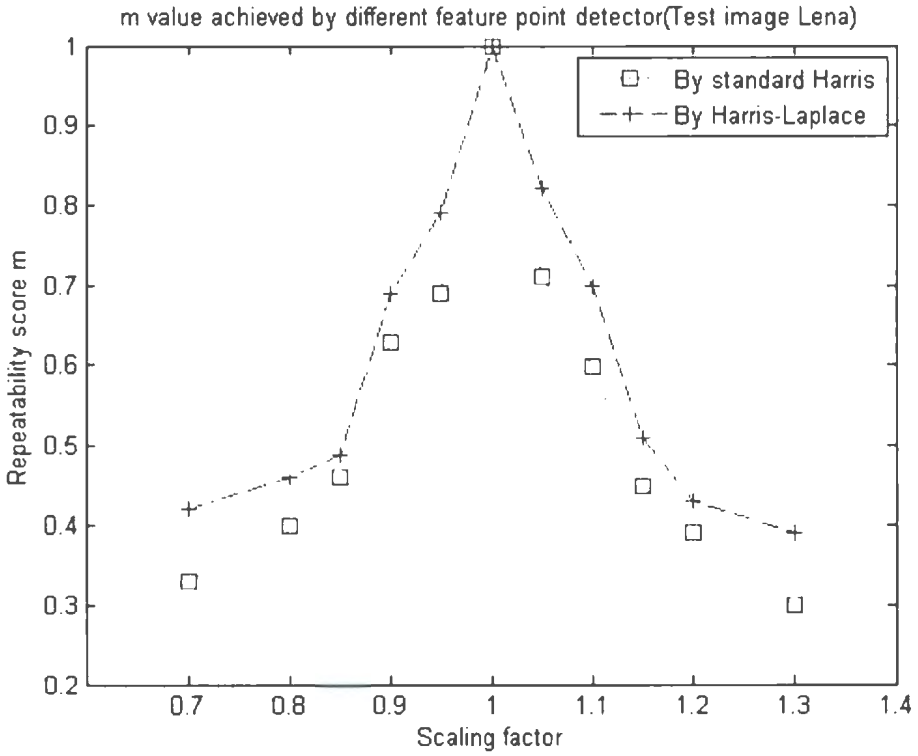
(a)Detecting feature points along four scales

(b)Feature points existing at the same location but different scales are preserved.

Figure 4.3.5: Select feature points existing in four scale spaces

To show the gain compared to the non-scale invariant method, parameter m in [84] is adopted to evaluate the capability to preserve the feature points for the standard Harris detector and the modified Harris-Laplacian detector. The parameter m is a repeatability score, computed as a ratio between the number of point-to-point

correspondences that can be established for detected points in two images and the number of feature points present in the original image: $m = N_{pre} / N_{ini}$, where N_{ini} denotes the number of feature points present in the initial image, and N_{pre} represents the number of preserved feature points after the image undergoes scaling transformation. When m is equal to 1, all of the feature points are preserved.



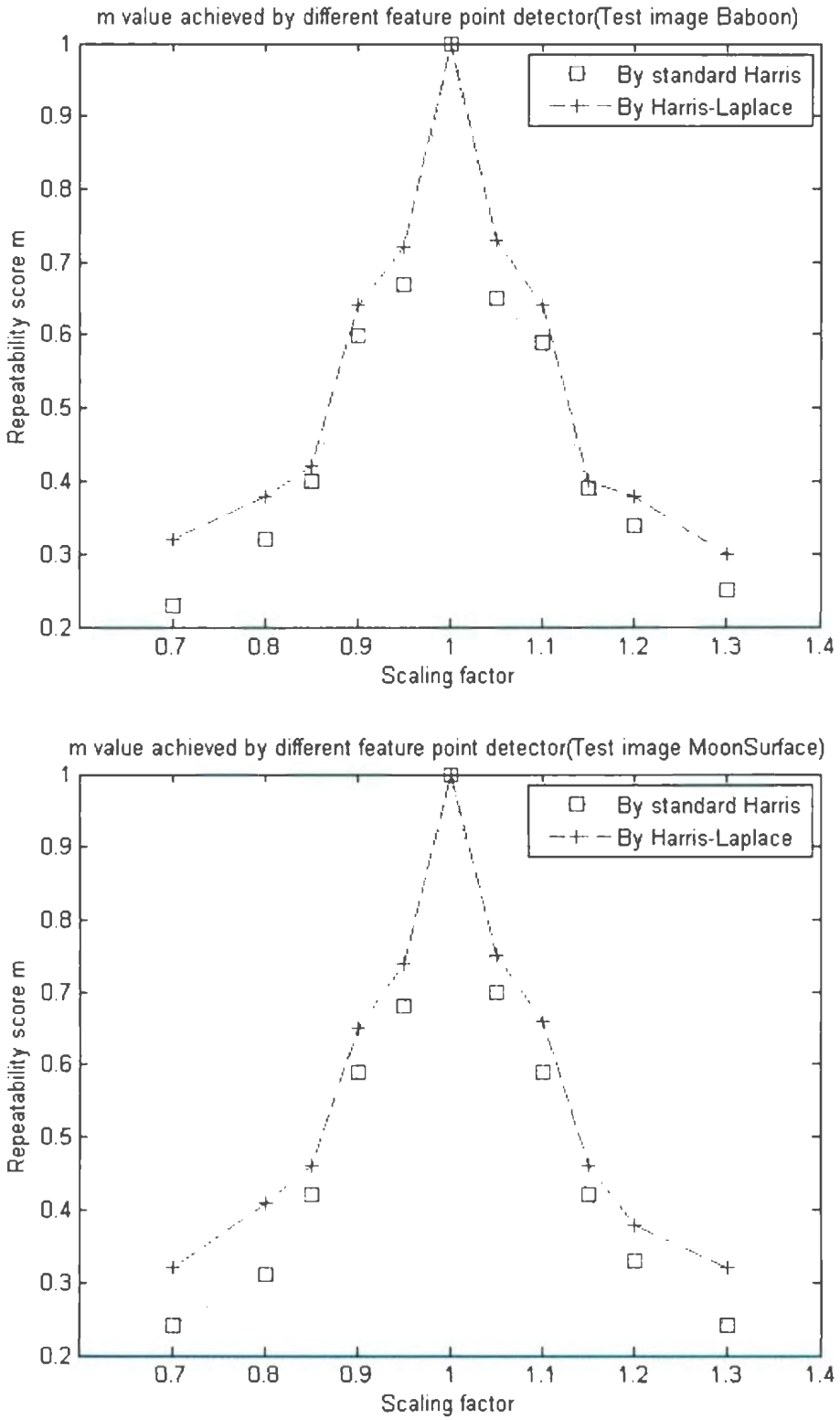


Figure 4.3.6: Repeatability score m after scaling: scaling factors are from 0.7 to 1.3

The experiments were done on 10 sequences of real images. In each tested image a set of feature points (set 1) is extracted. Ten different scaling operations (at scaling factors of 0.7, 0.8, 0.85, 0.9, 0.95, 1.05, 1.1, 1.15, 1.2, 1.3) are then performed and another set of feature points (set 2) extracted. The images then undergo an inverse transformation in order to compare feature points set 2 with set 1 at the same scale, so that the re-extraction rate m can be calculated. The experiments were done on 12 images (Appendix A) and the m values of three images (one for each testing class) are presented as examples in Figure 4.3.6. It is suggested that the Harris method modified with the scale space theory has higher re-extraction rate than the standard Harris method.

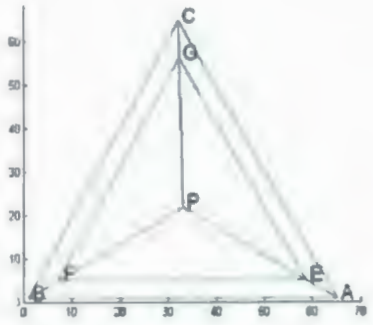
4.4 Embedding Unit Design

Two methods are commonly used for embedding unit design. In one method, several large regions are retrieved by adaptive K-mean feature points clustering. These regions are fit by ellipsoids and their bounding rectangles are used as embedding units. However, the effectiveness of this method is dependent on the contents, objects and textures of the image, and therefore it is not applicable to high textured images due to the difficulty in selecting appropriate regions. Furthermore, this method is sensitive to certain image processings such as cropping and filtering, which may alter the image contents. Another method, which is adopted in this work, uses Delaunay tessellation to decompose an image into a set of disjoint triangles. Delaunay tessellation is a fundamental computational geometry structure. For a given set of points, the Delaunay triangulation provides a set of lines connecting each point to its natural neighbors.

There are several reasons why Delaunay tessellation is used. First of all, the patches formed during Delaunay tessellation can spread throughout the image and do not overlap, and therefore the embedding scheme so established shows large capability and low bits error. Secondly, the tessellation has local properties: if a vertex disappears, the tessellation is only modified on connected triangles. Thirdly, each vertex is associated with a stable area. The tessellation is not modified when the vertex is moving inside this area.

The main consideration for embedding watermark into image units is the robustness of the image units. At the detection stage, feature points are expected to be robustly distributed at corresponding positions at which the feature points were extracted at the embedding stage. As a fact that the feature point detector is sensitive to even a small change of pixel value close to the feature points, feature points redetections could be affected by watermark insertion itself. Methods to prevent the loss of tessellation triangles have to be designed.

In this work, instead of using vertices of Delaunay tessellation triangles themselves, relative reference vertices are calculated in our scheme as shown in Figure 4.4.1(a). Let A, B, C be three vertices of a Delaunay tessellation triangle, P is the centroid of a triangle, the relative reference vertices are E, F, G, where $PE / PA = PG / PC = PF / PB = \alpha$, α is a predefined parameter.



(a) Modified Delaunay tessellation triangle

(b) Detected triangles of original image

(c) Triangles Redetection, Above "Without modification", Bottom "With modification"

Figure 4.4.1: Modification of the Delaunay triangles

Since extracted Delaunay tessellation triangles are shrunk moderately by introducing the parameter α , the robustness of feature point redetection is improved significantly by avoiding embedding a watermark at the location of corners. Moreover, the modified triangle method can avoid the superposition of signals between adjacent triangles which could cause severe problems when decoding the embedded information.

In the experiment, an α value of 0.9 is selected because it produces triangles with limited loss of capability while small enough for the triangles to avoid the location of feature points. Therefore, the α 0.9 is the tradeoff of the triangles embedding capability and robustness.

In order to demonstrate the superiority of modified triangle decomposition, a comparison is made with the original method under the same embedding strength. In Figure 4.4.1(c), the top three images show triangle redetection with Delauay tessellation and the bottom images show the redetection with modified Delaunay tessellation. The two images on the left present the watermarked images and the two in the middle present the embedded watermarks. The images on the right show the re-extracted Delaunay triangles. It is seen that several feature points are falsely detected in the scheme without modification, while all the feature points are preserved with the modified triangle embedding scheme. More examples of modified the triangle decomposition effects on re-extracting elementary triangles can be found at Appendix C.

4.5 Correlation-Based Multiple Bits Embedding Method

The extracted reference information is repeatedly embedded into each image unit (Delaunay triangle). Ignoring triangle warping, each image unit can be regarded as an individual cover image of small size. This section explains the mechanism of the basic correlation-based multiple bits embedding and extraction scheme. The overall embedding

and extracting scheme involving triangle warping is presented in details in the next section.

To design a robust and imperceptible multiple bit embedding scheme, techniques of spread spectrum communication are combined within watermark generation. To successfully decode the watermark, image restoration techniques are employed. Image restoration is used to obtain an approximate estimate of the original cover image from the watermarked image. This promotes the estimation of the embedded watermark that was added to the cover, in addition to allowing the scheme to be a blind watermarking scheme. Finally, because the watermark signal is of low power and the restoration process is not perfect, the estimate of the embedded signal could be poor, resulting in an embedded signal bit error rate (BER) that is rather high. To compensate, the watermark signal is encoded using a low-rate error-control code before embedding. This conglomeration of communication and image processing techniques provides a method of reliable blind image watermarking scheme.

4.5.1 Watermark Generation

The fundamental concept of watermarking is to embed information (watermark) into a digital cover image imperceptibly. The watermark signal can be treated as noise is the cover image, which should be imperceptible and undetectable. Spread spectrum techniques are adopted based on this requirement.

In spread spectrum communication, a narrow band signal is transmitted over a much larger bandwidth such that the signal energy present in any single frequency is undetectable. This can be accomplished by modulating the narrowband waveform with a

wideband waveform, such as white noise. In this work, the watermark is modulated with a binary pseudo-noise (PN) signal, since the binary pseudo-noise (PN) signal is wideband waveform and therefore the energy of its modulation with the watermark in any one frequency band is low and therefore difficult to detect [75]. Moreover, the watermark bits are spatially spread by a large factor cr (chip-rate) for additional redundancy. This spreading is accomplished by embedding one bit of the watermark into cr pixels of the image.

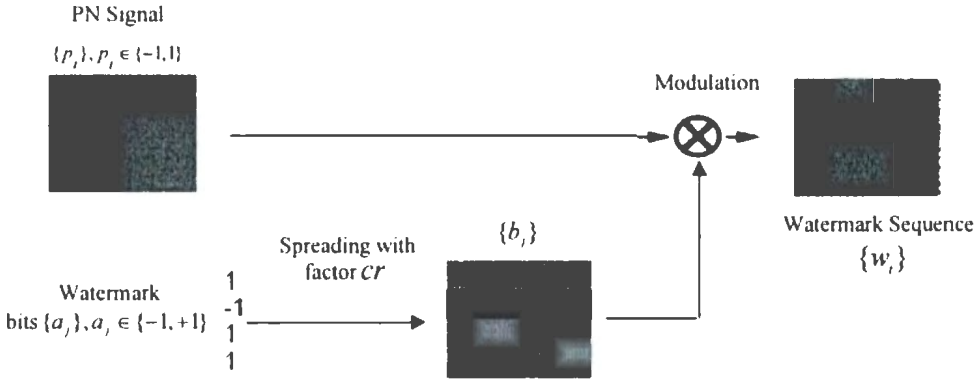


Figure 4.5.1: Spread spectrum watermark generation

In order to understand the effect of spread spectrum techniques, the general model for this method to derive the watermark sequence is shown in Figure 4.5.1. Let N be the total number of pixels in a cover image. Let cr be the chip-rate used to spread the information bits. Let L be the length of information bits which could be embedded in the cover image, and then the chip-rate $cr = N / L$. A sequence of information bits $\{a_j\}$, $a_j \in \{-1,1\}$ has to be embedded into the image. For example, in Figure 4.5.1, $\{a_j\}$ = [1 -1 1 1], in other words L =4, if given N=512, then cr = 512/4=128.

This discrete watermark signal $\{a_j\}$ is spread by the chip-rate cr to obtain the spread sequence $\{b_i\} : b_i = a_j$, where $j \cdot cr \le i < (j+1) \cdot cr$. The spread sequence $\{b_i\}$ is then modulated by a pseudo-noise (PN) sequence with N bits $\{p_i\}$, $p_i \in \{-1, 1\}$. This PN sequence $\{p_i\}$, serving for frequency spreading is generated by a wideband pseudorandom noise generator with a secret key K. This secret key is known by both the embedder and extractor. The modulated signal is scaled with a factor α:

$$w_i = \alpha \cdot b_i \cdot p_i$$

where w_i is the spread spectrum watermark bits, which is arranged into a matrix with size equal to the image size.

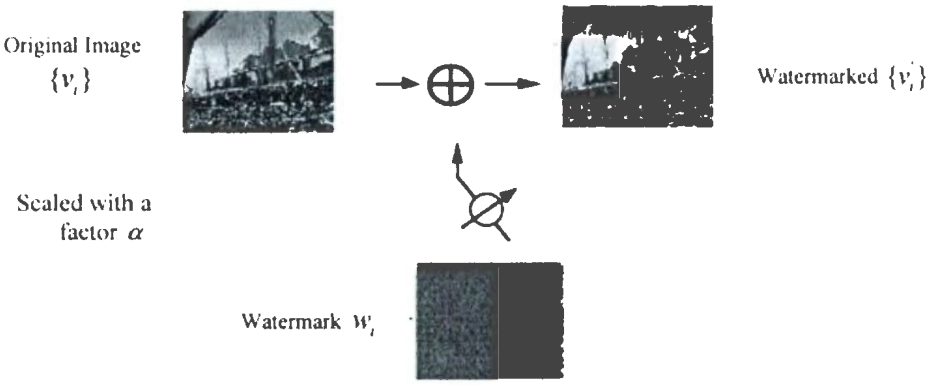


Figure 4.5.2: Watermark insertion

As shown in Figure 4.5.2, the spread spectrum watermark bits w_i, are added to the image pixel values, v_i, yielding a watermarked image, v_i':

$$v_i' = v_i + w_i.$$

4.5.2 Watermark Extraction

The watermarked image is then transmitted in some manner to the extractor, who maintains the same key K as the embedder. At the extractor, the embedded watermark must be extracted from the received watermarked image in order to be decoded. To do this, image restoration techniques, which filter most of the low-power embedded signal from the watermarked image, are implemented within the system to obtain an estimate of the original cover image. By subsequently subtracting the restored image from the received watermarked image, an estimate of the embedded signal is acquired.

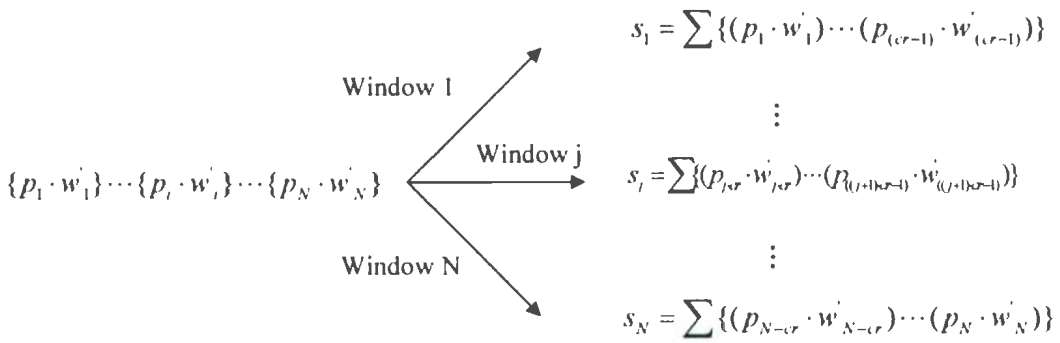
The general process of the spread spectrum extractor is described as shown in Figure 4.5.3. Assume that the watermarked image is $v_i' = v_i + w_i$, where w_i is embedded watermark. Filtering operations can be used to restore the original cover image v_i and restored image is expressed as $\hat{v}_i$. An estimate of the embedded signal w_i' is acquired by subtracting the restored cover image from the watermarked image:

$$w_i' = v_i' - \hat{v}_i \approx v_i' - v_i = w_i.$$

The restored image $\hat{v}_i$ can be obtained using a variety of image processing filters, such as mean and wiener filters. Effects of the wiener filter, a particular filter used in this scheme, on information extraction are demonstrated in the next section.

Once obtained from the image restoration, w_i' is then demodulated with an identical copy of the pseudorandom wideband sequence $\{p_i\}$ used at the embedder. The generation of $\{p_i\}$ is accomplished by the possession of a common key, which is used as

a seed for duplicate random number generators. The two signals are modulated $\{p_i \cdot w'_i\}$, and summed over a window of length equal to the chip rate cr yielding the correlation sum s_j for the j^{th} watermark bit as depicted below:



If the estimated watermark signal w'_i can be approximated by w_i, the above summation function can deduce following formulas:

$$s_j = \sum_{i=jcr}^{(j+1)cr-1} p_i \cdot w'_i \approx \sum_{i=jcr}^{(j+1)cr-1} p_i \cdot w_i = \sum_{i=jcr}^{(j+1)cr-1} p_i^2 \cdot \alpha \cdot b_i \ ,$$

$$s_j \approx cr \cdot \alpha \cdot b_i = cr \cdot \alpha \cdot a_j .$$

Because $r_i > 0, \alpha > 0, p_i^2 = 1$ and $a_j = \pm 1$, embedded bits can be retrieved:

$$sign(s_j) = sign(a_j \cdot cr \cdot \alpha) = sign(a_j) = a_j .$$

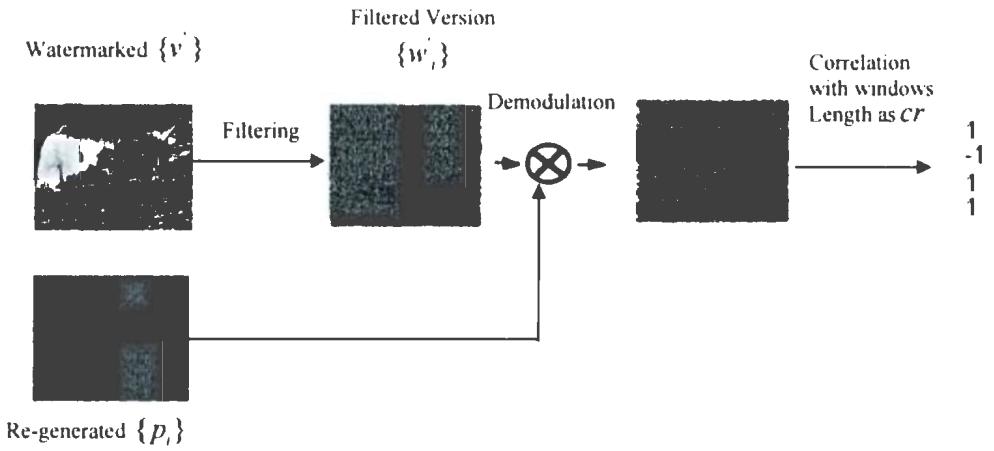


Figure 4.5.3: Extracting stage

For each summation window, the extracted watermark bit is 1 if the summation is positive and -1 if it is negative. The retrieval scheme is shown in Figure 4.5.3.

4.5.3 Adaptive Noise-Removal Filtering

To improve watermark extraction, the major part of the cover image is estimated and removed from the watermarked image. Filtering can be considered as a denoising operation and allows separation of the image components from the mark components. As stated earlier, the restored image can be obtained with a variety of image processing filters.

The Wiener filter in Lee's algorithm [85] is a good option for adaptive noise-removal filtering and is adopted in this work. It gives a low overall mean-squared error (MSE) between the filtered image and the watermarked image, thus providing a restored cover image that is much like the original cover image. Wiener adaptive noise-removal filter uses a pixelwise adaptive wiener method, based on statistic estimating from a local

neighborhood of each pixel. The Wiener filter estimates the local mean and variance around each pixel,

$$\mu = 1/(N \cdot M) \sum_{(n_1, n_2) \in \eta} v'(n_1, n_2)$$

$$\sigma^2 = 1/(N \cdot M) \sum_{(n_1, n_2) \in \eta} v'^2(n_1, n_2) - \mu^2$$

where $v'(n_1, n_2)$ is the watermarked image that has been degraded by constant power additive noise (watermark), η is the N-by-M local neighborhood of each pixel in the image $v'(n_1, n_2)$. Based on these pixelwise estimations, an estimated image $\hat{v}(n_1, n_2)$ can be achieved by

$$\hat{v}(n_1, n_2) = \left\lfloor \mu + \frac{(\sigma^2 - \delta^2)}{\sigma^2} \cdot (v'(n_1, n_2) - \mu) \right\rfloor$$

where δ^2 is the noise variance which is approximated by the average of all local estimated variances.

An example of effects of the Wiener filter is given below. One information bit $\{a\} = \{-1\}$ is going to be embedded into following cover image v (presented as matrix, size as 10* 10):

175	133	156	156	156	154	156	175	164	147
137	129	153	158	192	211	149	166	174	105
135	130	192	167	196	219	225	189	103	185
145	133	200	168	127	106	215	127	105	180
130	133	147	94	164	194	221	109	185	184
113	135	172	97	129	183	159	96	178	181
130	131	163	143	150	176	163	106	177	216
141	146	143	113	105	178	111	177	174	214
123	114	105	190	155	167	212	176	218	146
200	101	104	96	155	164	190	140	153	152

Firstly, a pseudo random sequence, $P = \{p_i\}, p_i \in \{-1, 1\}, i = 1 \cdots 100$, is generated by key $K = 25$. Then this pseudo random sequence P is modulated with information bit $\{a\}$ = {-1}, and is embedded into the cover image v. The watermarked image v' is derived as:

176	132	157	155	155	153	157	176	165	148
136	128	154	159	191	212	150	165	175	104
134	129	193	166	197	220	226	190	102	184
144	134	199	169	128	105	216	126	104	181
131	132	148	95	163	193	222	110	184	185
112	134	171	96	128	182	160	95	177	182
131	130	162	142	151	175	162	107	178	217
140	145	142	112	104	179	112	178	173	215
122	113	106	191	154	166	213	175	219	147
199	100	103	95	156	163	191	139	152	153

At the detection stage, watermarked image v' is filtered with the Wiener filter to get the estimated cover image $\hat{v}_i$:

117	113	124	133	138	137	138	146	136	108
110	149	153	170	179	184	183	167	157	101
104	150	159	173	172	183	179	161	148	138
107	149	152	162	159	186	179	164	152	148
100	145	142	144	140	166	156	155	149	156
92	139	134	139	147	171	156	155	159	162
101	141	137	134	141	150	150	149	169	187
102	132	138	141	153	157	163	168	179	187
104	130	123	129	147	160	168	172	172	136
123	89	85	92	131	145	162	132	134	124

Since the Wiener filter can not completely remove all the embedded watermarks, the estimated cover image $\hat{v}_i$ differs to some extent from the original cover image. The estimated signal w'_i is achieved by watermarked matrix subtracting the filtered matrix $w'_i = v' - \hat{v}_i$:

59	19	33	22	17	16	18	30	29	40
27	-21	1	-11	12	27	-33	-2	18	3
30	-21	34	-7	26	37	48	28	-45	45
38	-16	47	7	-32	-81	37	-39	-48	33
31	-13	6	-49	23	27	66	-45	35	29
19	-5	37	-44	-19	11	4	-60	18	20
30	-11	25	7	10	25	12	-42	9	30
38	12	4	-28	-49	22	-51	9	-5	28
18	-17	-17	62	7	6	45	2	46	11
76	12	17	3	25	19	29	6	18	30

Finally, the estimated signal w_i' is modulated with the original pseudo random sequence P. The summation of the modulated values is -5.0868. So the extracted embedded bit can be classified as -1, which is the same as the embedded information bit $\{a\}$ =-1.

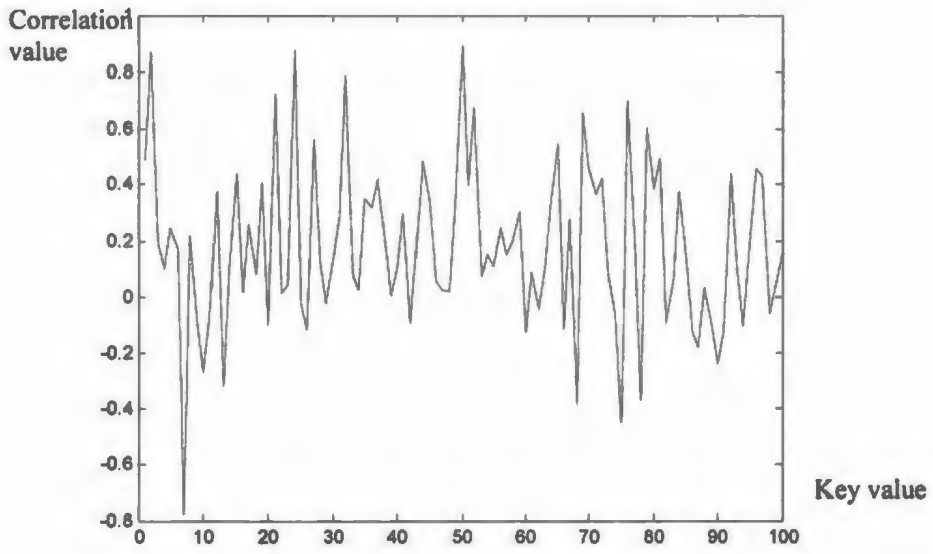


Figure 4.5.4: Extraction results without Wiener filter

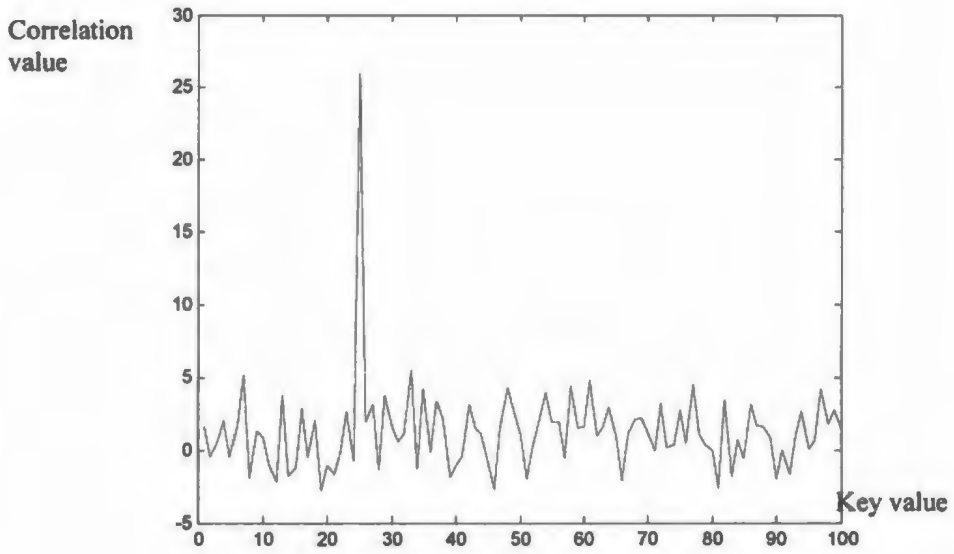


Figure 4.5.5: Results after wiener filter, key=25

Another example is given with the embedded information as $\{a\} = \{+1\}$. We can generate 100 different random sequences with 100 different keys. The random sequence generated by key $K = 25$ (x axis) is embedded into the cover matrix. Of the 100 random sequences tested, only the sequence that was originally embedded should yield a high correlation output at the detection stage, because the same random sequence as the one used in embedding stage can be regenerated by $K = 25$. Effects of Wiener filtering are represented by Figure 4.5.4 and Figure 4.5.5. The results indicate that Wiener filtering enables the detection of the embedded bits, i.e. the correlation value reaches a peak at key of value of 25, whereas using classical correlation detection alone can not give any distinguishable peak.

4.5.4 Error-Control Coding

Since image restoration does not result in a perfect copy of the original cover image and the embedded watermark signal is low power, the estimate of the embedded watermark signal could be poor. The result of the demodulated signal may have a substantial number of bit errors, indicated by a high embedded signal bit error rate. Therefore, to allow for suboptimal performance of the signal estimation process, the use of low-rate error-control codes is adopted to correct bit errors.

Any error-correcting code that is capable of correcting the high bit error rate can be used. Among various error-correcting coding algorithms, a decoder based on convolutional coding and soft-decision Viterbi decoding algorithm [20] is selected in this scheme for error correction due to several reasons. First of all, the watermarking scheme with low signal-to-noise ratio (SNR) can be considered as a noisy channel in the communication theory, while the watermark can be considered as an additive white gauassian noise (AWGN). Therefore, the convolutional coding with soft-decision Veterbi decoding algorithm, which has strong correcting capability towards AWGN, is suitable for the proposed watermarking scheme. Secondly, since the sizes of patches in content-based watermarking schemes are relatively small, the system requires good performances of the decoding algorithm at low implementation cost. Among the existing coding and decoding algorithm in watermarking applications, it has been that the combination of convolutional encoding and soft-decision Viterbi decoding serves as an effective error-correcting coding for content-based watermarking schemes [77].

4.5.5 Triangle Warping

The extracted reference information is spread into a standard triangle by the basic correlation-based multiple bits embedding and extraction scheme described above. Then the standard triangle is warped to fit into each extracted Delaunay triangle. Warping a triangle into another triangle can be done via affine transformation, followed by cubic-spline interpolation. The cubic-spline interpolation preserves high frequencies of image signals, where embedded watermarks are located.

In order to warp a Delaunay triangle T_D into a standard triangle T_S, an affine transformation A that satisfies the mapping of every vertex (x_s, y_s) in T_S into its corresponding vertex (x_d, y_d) in T_D is calculated. Since the angles are descending, so the transformation is unique as shown in Figure 4.5.6. The affine transformation A is expressed as the following equation and the six real parameters a, b, c, d, e, f are obtained by inserting three pairs of known vertices.

$$A(x_s, y_s) = \begin{pmatrix} x_d \\ y_d \end{pmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{pmatrix} x_s \\ y_s \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}.$$

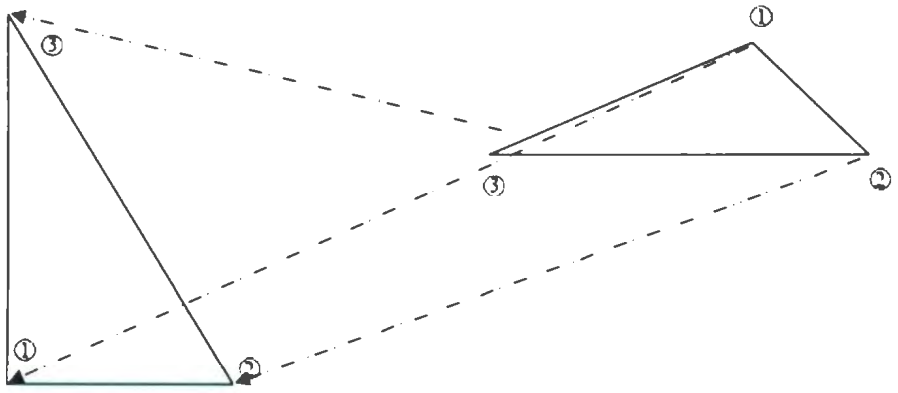


Figure 4.5.6: Orientation of the triangles

This affine transform A permits us to model geometric transformation, such as rotations, scaling operations, or shearing effects and also allows us to transform any point of the triangle T_s into a corresponding point of the triangles T_D . A cubic-spline interpolation process is applied on the neighborhood of the affine point to obtain the pixel value.

Similarly, the six parameters corresponding to the affine transformation A^{-1}, a process that maps the vertexes of the standard triangle into those of the Delaunay triangle, can be computed.

4.6 Embedding Scheme

After extraction of Delaunay triangles and reference information σ_o and θ^o, the reference information is merged into each Delaunay tessellation triangle. The whole embedding scheme is given in Figure 4.6.1 and the main formatting and merging steps are summarized as follows:

Step1: Reference information σ_o and θ^o are represented by N_b -bit binary antipodal vector $b = (b_1, ..., b_{N_b}), b_i \in \{-1, 1\}, \forall i \in \{1, ..., N_b\}$.

Step2: Vector b is encoded by a convolutional encoder, resulting in the N_c -bit antipodal coded vector c .

Step3: Vector c is then duplicated into a N_c'-bit vector c', where $N_c' = cr * N_c$ and cr is called the expansion window or the spread factor.

Step 4: By a pseudo random sequence $p = (p_1, ... p_{N_c'})$, $p_i = \{-1, 1\}$,

$\forall i \in \{1, ..., N_c'\}$, c' is modulated into vector $s : s_i = p_i \cdot c_i'$. s is the signal actually embedded.

Step 5: A standard triangle T_s is defined as a 96*96 isosceles right-triangle. The 1-D sequence s is arranged into the shape of a standard triangle T_s.

Step 6: the standard triangle T_s is warped into the shape of each Delaunay triangle T_D to obtain T_{map}.

Step 7: T_{map} and T_D are added together to obtain a marked triangle with an embedding strength and perceptual mask as presented in [77].

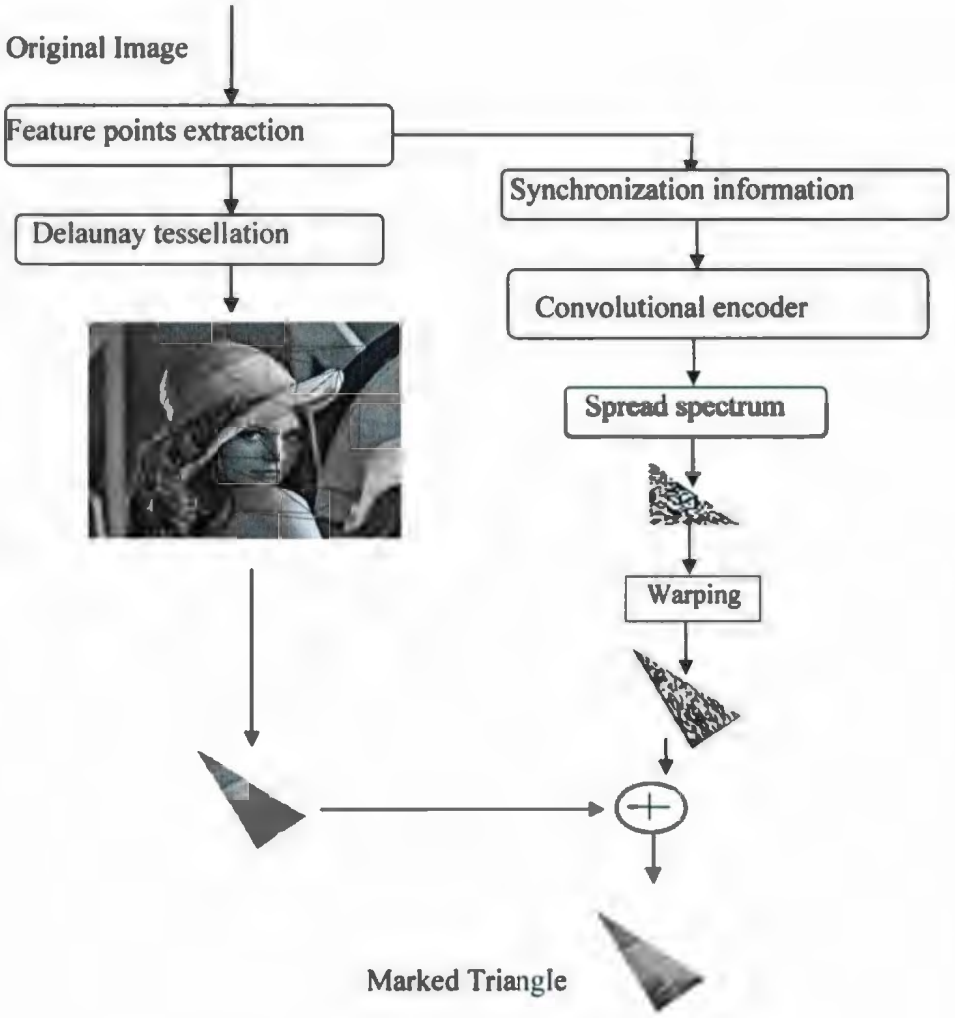


Figure 4.6.1: Embedding scheme

4.7 Extracting Scheme

The detection scheme is shown in Figure 4.7.1. Similar to the embedding stage, the Delaunay triangles and synchronization information (σ_s , θ') are redetected, and the steps of extraction of embedded information, σ_o and θ'' from each Delaunay triangle can be summarized as following:

Step 1: The redetected T_D is warped into the shape of standard triangle T_s.

Step 2: The pre-defined Wiener filter is applied to filter much of the low-power embedded signal to estimate the original message. By subsequently subtracting the restored image from the received embedded image, an estimate of the embedded signal s is acquired.

Step 3: The noise sequence p is regenerated.

Step 4: c is estimated from the formula:

$$\sum_{t=k\cdot cr}^{(k+1)\cdot cr-1}(p_t \cdot s_t) - (\sum_{t=k\cdot cr}^{(k+1)\cdot cr-1} p_t)\cdot E(\sum_{t=k\cdot cr}^{(k+1)\cdot cr-1} s_t) = cr \cdot c_k \,, \forall k \in \{1,...,N_c\}\,.$$

Step 5: The embedded message σ_o and θ'' is decoded by Viterbi soft-decision [76].

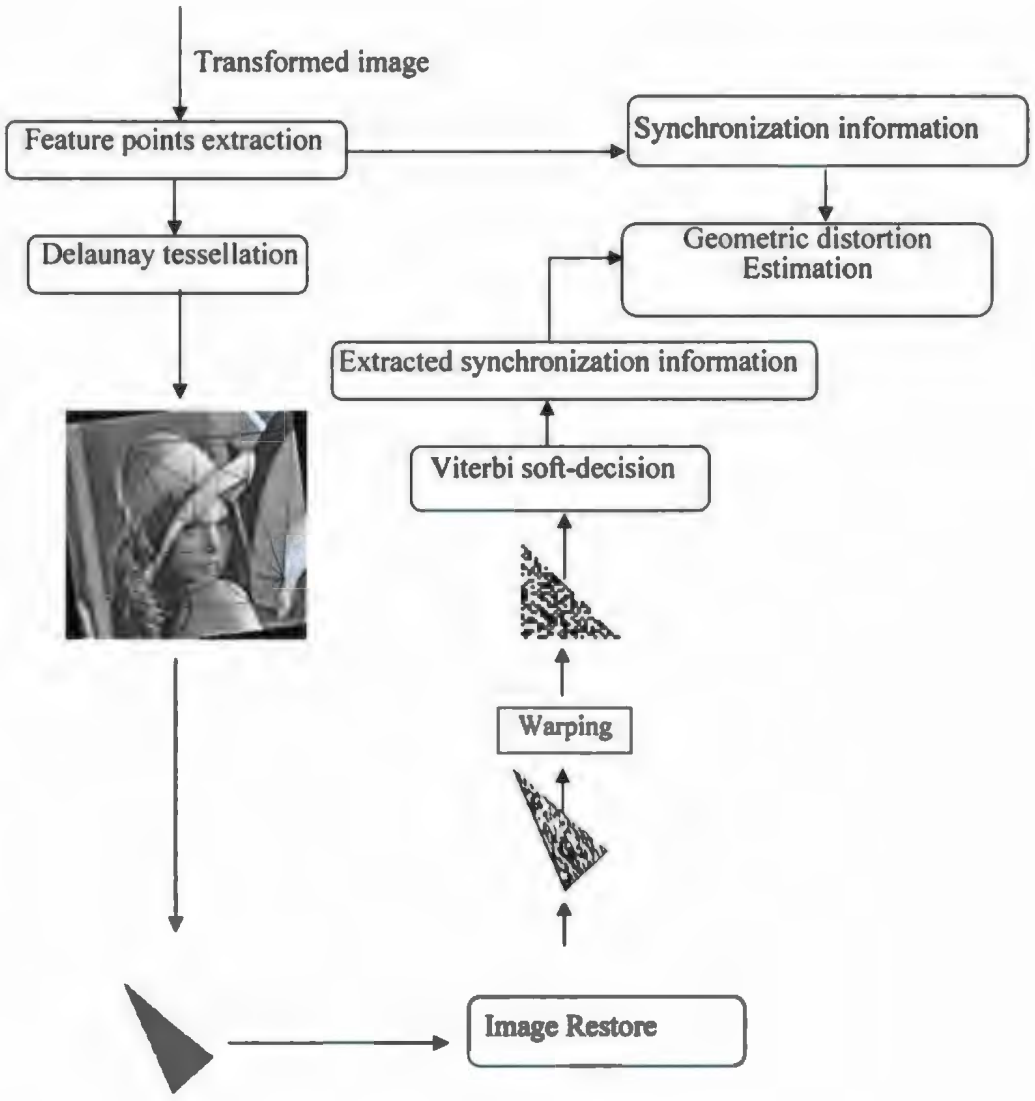


Figure 4.7.1: Extracting scheme

Chapter 5

Performance and Discussion

The objective of this chapter is to show the robustness of the proposed scheme against rotation and scaling distortion. The experimental set-up for the scheme is discussed and the results from our implementation and its performance analysis are presented. The factors that seem to affect the performance of the scheme are also discussed. Furthermore, some theoretical and practical limitations of our scheme are presented in this chapter as well.

5.1 Experiment Procedure

The tests were processed on 12 different images representing different classes of contents. The test database is shown in Appendix A. Various categories of images have been chosen. Some images (Lena, Man, Couple, Pentagon) include very distinctive

corners, and other images (Baboon, Boat, Aerial, Bridge) have textured areas with high frequency components while the other images (MoonSurface, Car, Hare, Peppers) include large homogeneous areas.

The general framework of our scheme can be summarized as three basic steps. First, for each test image, a set of Delaunay triangles is formed by feature points. Next, watermarks with the same embedding weight factor, generated from reference information for each test image, are embedded into the test image by the proposed feature based watermarking method. Then, a sequence of geometric distortion, scaling and rotation, is applied to each watermarked image. At the detection stage, the reference information is extracted from each Delaunay triangle of the sequence of distorted images. The number of triangles in which the reference information can be successfully extracted, reflects the robustness of the data hiding scheme.

This fact that our geometry-invariant watermarking scheme can be functionally partitioned into three modular blocks allows independent evaluation of each essential building block. In addition, the partitioning also allows us to analyze the requirements for each component individually. First, the precision of geometric distortion estimation based by the reference location information has to be guaranteed. Second, the extraction of elementary triangles has to be robust. Finally, the capability of content-based watermarking scheme must be high enough to embed the location reference information.

In following sections, performance of the proposed scheme on each module is presented, and then a discussion of the requirements on each block is followed by analysis of the presented scheme to meet these requirements.

5.2 Geometric Distortion Estimation

In this step, geometric distortion is estimated by comparing the embedded location reference information (feature points location of original images) with the actual location information (feature point location of transformed images). Feature points are extracted by scale-space adapted Harris corner detection. The performance and a discussion of the requirements on this geometric distortion estimator are presented in this section.

5.2.1 Rotation and Scaling Estimation Results

The scaling factor is approximated by the Edges Standard Deviation Ratio (ESDR) which is achieved by comparing the deviation of the feature point location before and after the attack has been performed. The rotation angle is approximated by Average Edge Angle Difference (AEAD).

Figure 5.2.1 shows an example of the extracted feature points from an original image and the image with scaling factor 0.8. Figure 5.2.2 illustrates the locations of extracted feature points from an original image and the image with a rotation factor of 5 degrees. The scaling and rotation estimation can be achieved by comparing reference locations of these feature points. The location of most red dots in distorted images (b), which represent the extracted feature points, remains unchanged compared to those in the original images (a).

The robustness of our geometric distortion estimator is evaluated by testing on the testing image database. For example, a scaling operation of 120% and rotation of 5

degree are performed on our testing image database. Figure 5.2.3 (a) and Figure 5.2.3 (b) show the estimated scale factor and the estimated rotated angle for each image, respectively. The estimated distortion factors of all tested images, given as circles, appear close to the actual factors of 1.2 and 5, respectively, for scaling factor and rotation angle.

Figure 5.2.1 (a) Original image (b) Image scaled 0.8

Figure 5.2.2: (a) Original image (b) image rotated 5 degree

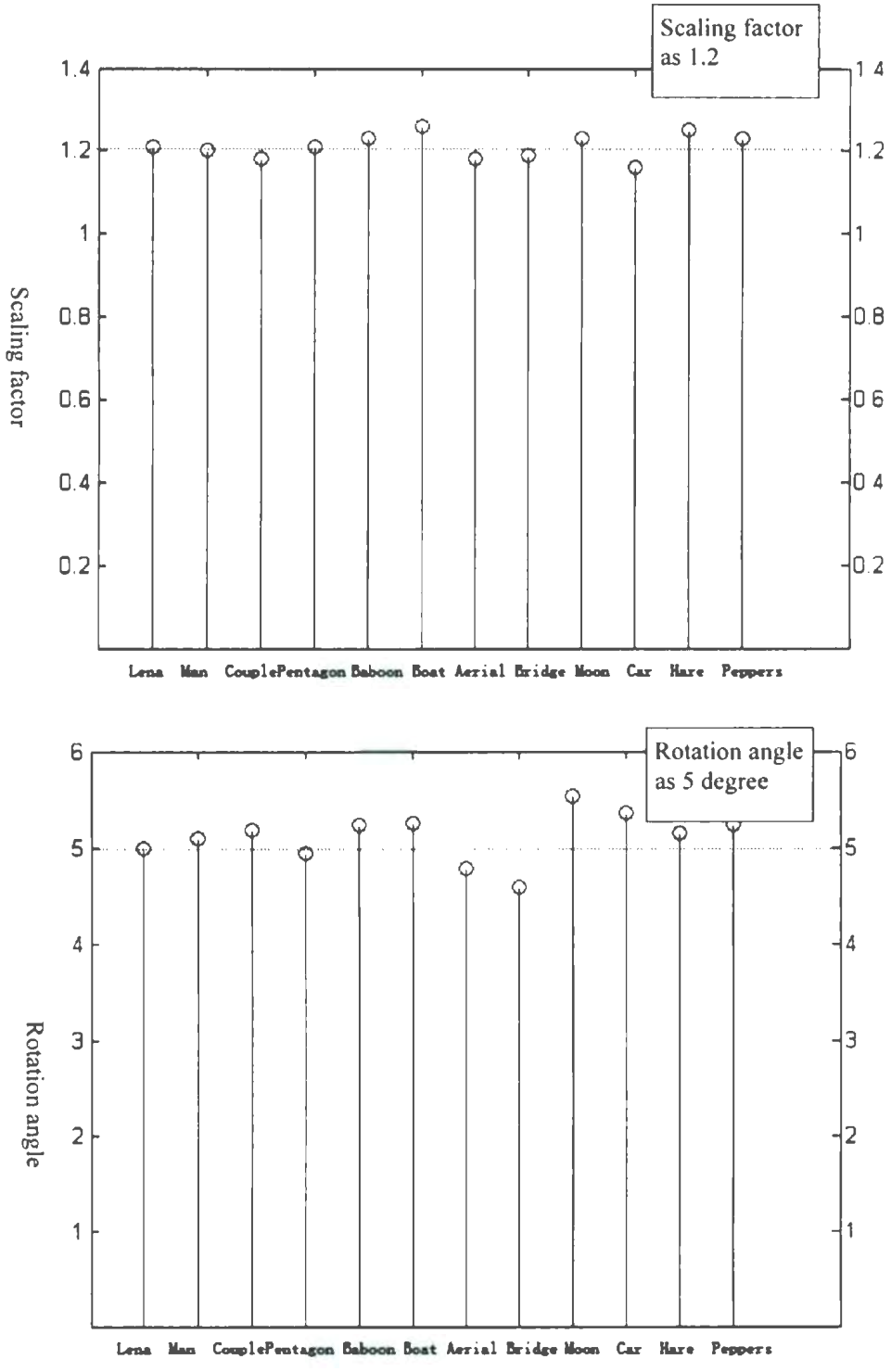


Figure 5.2.3: Estimation results of 13 testing images

As stated earlier, the testing image database is classified into three categories, each containing four images with similar characteristics. Each image is analyzed for its Squared Error (SE) at scaling factors ranging from 0.7 to 1.3 and rotation angles ranging from - 15° to 15° as follows:

$$SE = (V_e - V_a)^2 ,$$

where V_a is the value of the actual distortion factor, i.e. scaling factor, and V_e is the value of the estimated distortion factor. Means Squared Error (MSE) is then obtained by averaging SE of four images for each category and shown in Figure 5.2.4.

The lowest MSE value is observed when the scaling factor is close to 1. MSE value increases with increasing scaling distortion, which is defined as the degree to which the image is rescaled. The first class has the lowest MSE, thus providing the best estimate of geometric distortion; whereas the second and third classes do not differ significantly.

Rotation distortions exert similar effects on MSE, i.e. the smaller the rotation distortion angle, lower the MSE value is. It is noticeable that when the rotation angle is in the range from -5 to 5, MSE is the lowest for all categories. During this range, the first class provides the best estimate among all categories. At higher rotation angles (<-5 or >5), MSE increases remarkably and shows no significant different among the three categories.

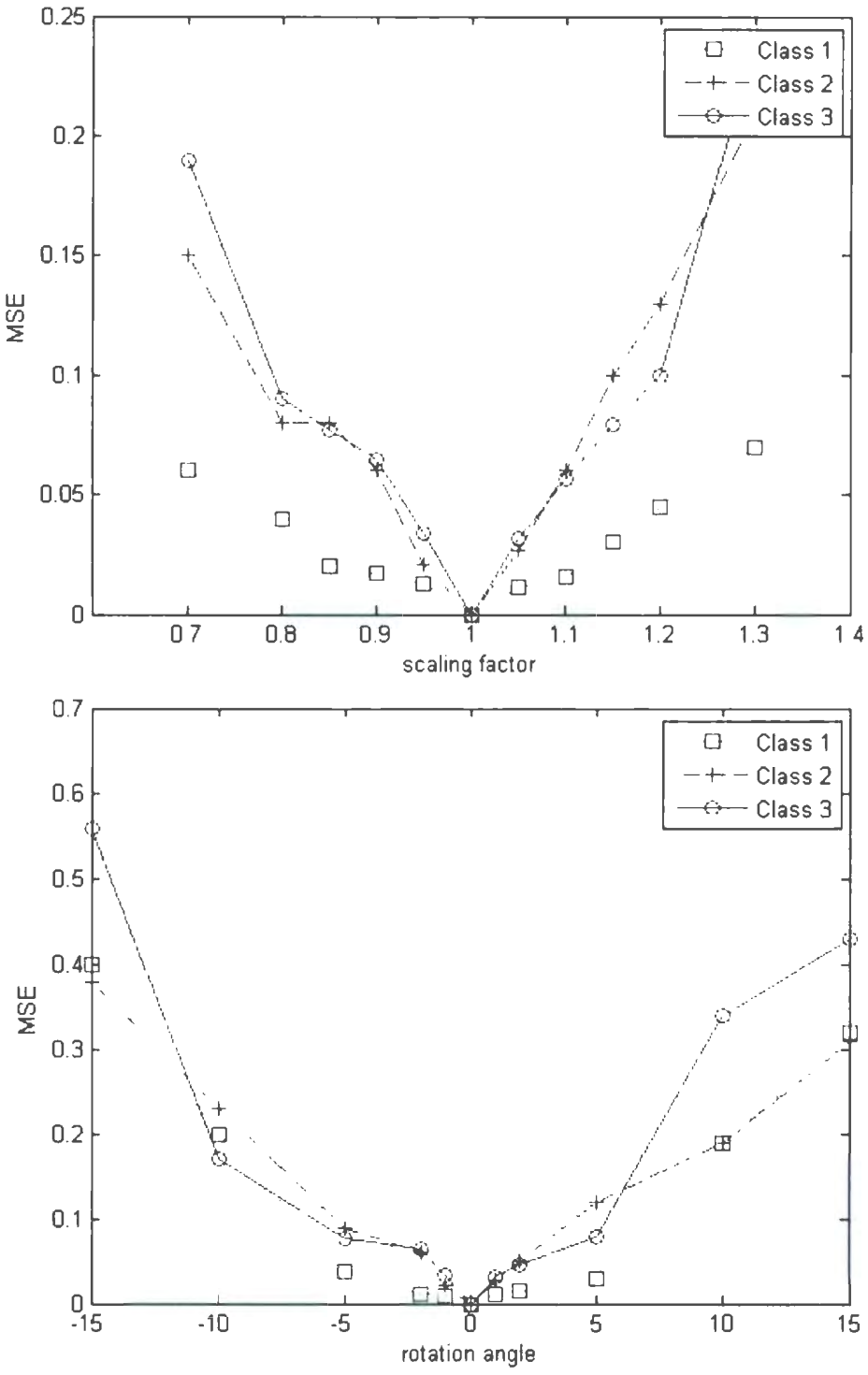


Figure 5.2.4: Estimation MSE value of different geometric distortions

As indicated from experimental results, the geometric distortion estimator can achieve promising results in a certain range of rotation and scaling distortion. Several important elements can also be deduced from experiments:

1) With increasing scaling and rotation distortion, the accuracy of geometric distortion estimation decreases rapidly. This is caused by significant loss of the embedded feature points.

2) The estimated rotation angle is accurate in a relatively small range due to the fact that the rotation factor is only estimated by average angle of feature points in the first quarter.

3) Images with distinctive corners (the first class) have the best estimate of the geometric distortion, while the estimate of scaling and rotation factor for images with highly textured areas or large smooth areas is less accurate. This is mainly due to the fact that feature points in such image areas can be significantly missing after undergoing geometric distortion.

5.2.2 Requirements of Proposed Scheme

This section discusses whether the accuracy of the geometric distortion estimation meets the requirements of the scheme. The results indicate that geometric distortion estimate has satisfactory performance under a certain range of scaling and rotation. Rotations at larger angles lead to reduced accuracy of the geometric distortion estimation, which depends on the statistical distribution of feature points in the first quarter; alteration of the distribution of feature points occurs during rotation at larger angles. Therefore, this

geometric distortion estimator is more applicable to data hiding schemes in which large rotation is not commonly seen.

Data hiding requires the maximum amount of data to be embedded invisibly into a cover image. Little research has been carried out to investigate the basic robustness against geometric distortions in data hiding schemes. Generally, higher capacity (the amount of information being embedded) is associated with lower watermark robustness. While many studies focus on the capacity, watermark robustness of the data hiding scheme against geometric distortion has been ignored. It has been assumed that data hiding schemes do not require high robustness. However, distortions such as resizing and rotation frequently occur during image manipulations and adversely affect the extraction of embedded data. This makes it difficult to the design of a data hiding scheme robust against geometric distortions while keeping high capacity.

This work aims to improve the robustness against geometric attacks in data hiding applications, where the geometric distortions caused accidentally by image manipulations are fine scaling and slight rotation. This geometric distortion estimation method provides good accuracy in the case of slight distortion and therefore offers a possible solution to counter geometric distortion in data hiding application.

5.3 Robustness of Extracted Element Triangles

After testing the geometric distortion estimator, the robustness of the element triangle extractor is evaluated in this section. As mentioned in the last chapter, since extracted feature points are bound to the image content, this set of points is used to divide the image into triangle patches, which are warped into a standard geometry ensuring the exact synchronization during insertion and extraction. The focus in the analysis of the requirements for this module is primarily on the repeatability of the image tessellation into elementary triangles.

5.3.1 Experimental Set-up of the Feature Point Extractor

The feature point extractor provides a large set of candidate points from which a smaller subset is selected as feature points based on the strength of the detector response. To obtain a homogeneous distribution of feature points in an image, a common technique is to select local maxima of the edge/corner-detector responses in a defined local neighborhood for each feature point. It is important to define the size of the neighborhood. If this size is too small, the distribution of the different feature points is concentrated on textured areas. If the size of neighborhood is too large, the feature points become isolated.

In this work, a circular neighborhood is used to avoid increasing detector anisotropy. The center of the neighborhood is the considered pixel. To be robust to scaling operations, the circle diameter depends on the image dimensions: $D = \frac{w+h}{\gamma}$. The integers w and h represent respectively the image width and height. The neighborhood size is quantized by the γ value.

(a) γ=100 (b) γ =30 (c) γ=10

Figure 5.3.1: Triangle distribution controlled by γ

In order to investigate the influence of parameter γ on the formation of elementary triangles, different γ values are used in the feature point extractor. Higher γ value results in a higher number of extracted elementary triangles, which is not desired in this case owing to the reduced size of triangles and thus deficient embedding reference information. However, the γ value should be high enough to obtain a proper number of triangles for reference information embedding. Among different γ values used for the 12 tested images in this work a homogeneous repartition is achieved near the γ value of 30, where a set of well distributed elementary triangles with reasonable sizes is obtained. Three sets of elementary triangles extracted from the image "baboon" at γ values of 100 30 and 10, for example, are demonstrated in Figure 5.3.1.

5.3.2 Performance of the Elementary Triangles Extraction

The most important criterion of the elementary triangle extraction module is the repeatability rate after undergoing rotation and scaling. The feature point extraction module extracts synchronization markers which are robust to geometric manipulations and to other content-preserving signal processing operations. It is a challenging task to find feature point extractors which produce repeatable results under the broad range of image processing operations.

In this test, geometric distortion attacks (rotation, scaling) are also applied on the 12 selected test images. As an example, the set of extracted elementary triangles of the "man" image under several different geometric distortions is shown in Figure 5.3.2. More examples of testing the robustness of elementary triangle extractor can be found at Appendix B. If the difference between the patches from the original image and the patches from the attacked images was less than two pixels, the patches are regarded as having been correctly redetected. These small misalignments can be compensated by searching some pixels around position of the patches originally found during watermark detection. In particular, prior to comparison, we reversed the coordinates of the patches in the attacked images into coordinates in the original image by calculating their inverse transform.

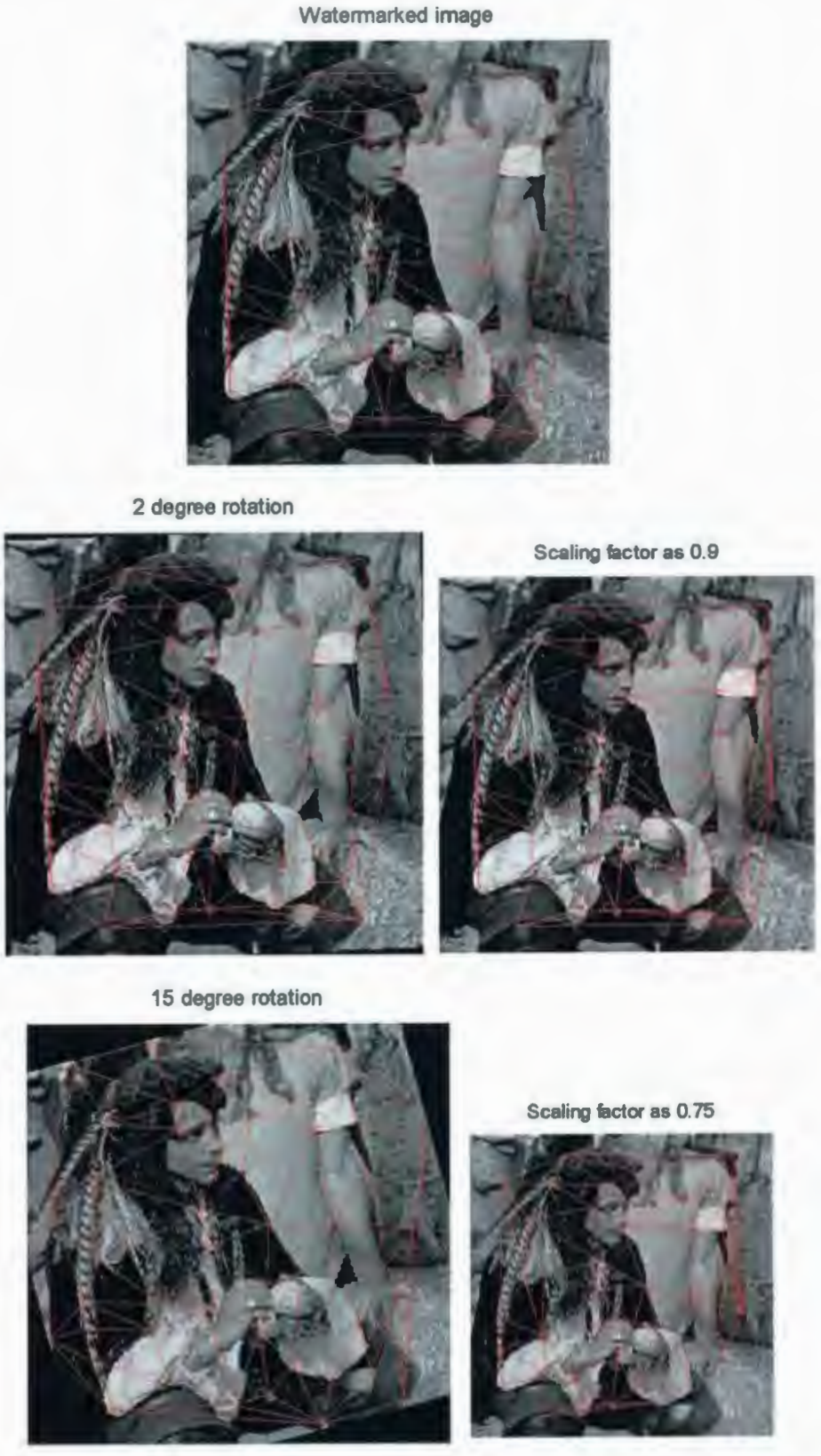


Figure 5.3.2:Extracted elementary triangles of different attacks

Table 5.3.1: Redetection ratios (%) under rotation distortions

	First Class				*Second Class*				*Third Class*			
	Lena	Man	Couple	Pentagon	Baboon	Boat	Aerial	Bridge	Moon	Hare	Car	Peppers
Rotated 1°	85	81	77	76	70	69	65	72	61	61	58	54
Rotated -1°	89	76	70	68	65	60	61	69	65	48	51	58
Rotated 2°	80	73	66	64	60	58	60	61	60	59	54	51
Rotated -2°	77	69	57	59	56	54	63	56	57	57	48	45
Rotated 5°	44	43	44	49	35	33	36	29	32	31	30	29
Rotated -5°	47	41	40	41	34	32	31	27	28	29	29	21
Rotated 10°	31	26	27	26	18	17	14	11	15	10	13	12
Rotated -10°	29	28	24	22	16	15	16	12	14	13	12	11
Rotated 25°	22	19	22	18	12	11	10	8	12	9	11	10
Rotated -25°	24	18	23	18	11	13	11	10	11	10	9	11

Table 5.3.2: Redetection ratios (%) under scaling distortions

	First Class				*Second Class*				*Third Class*			
	Lena	Man	Couple	Pentagon	Baboon	Boat	Aerial	Bridge	Moon	Hare	Car	Peppers
Scaling 0.95	71	69	67	60	53	58	51	50	50	52	38	53
Scaling 1.05	69	70	68	63	51	54	53	44	49	55	49	48
Scaling 0.9	58	59	57	50	48	51	48	46	46	47	37	45
Scaling 1.1	56	51	55	53	51	47	43	47	43	39	38	43
Scaling 0.85	37	32	40	36	27	20	21	22	21	25	23	17
Scaling 1.15	25	23	26	25	15	13	14	11	13	12	11	15
Scaling 0.8	19	22	21	20	14	11	12	10	11	10	9	12
Scaling 1.2	21	19	21	22	15	13	11	12	10	11	9	11
Scaling 0.7	13	16	15	19	14	12	10	9	7	10	7	8
Scaling 1.3	14	17	14	17	12	13	11	10	8	9	5	9

Results are expressed as the redetection ratio of triangles, which is the ratio of the triangle number detected in the distorted image to that in the original image. The distortions include ten different rotation angles (-25, -10, -5, -2, -1, 1, 2, 5, 10, 25) and ten scaling factors (0.7, 0.8, 0.85, 0.9, 0.95, 1.05, 1.1, 1.15, 1.2, 1.3), and the resultant redetection ratios are presented in Table 5.3.1 and Table 5.3.2, respectively, for varied rotation angles and scaling factors. Consistent with the previous evaluation of geometric distortion estimator, the test image database is grouped into the same classes: images with distinctive corners (first class), images with highly textured areas (second class) and images with large smooth areas (third class).

The results suggest that robustness of the extracted elementary triangles depends on the content of the images. The first class has the highest redetection ratio because the images contain sharp corners which contribute to robustness against geometric distortions. Lower redetection ratios are observed in the second class (textured images) and the third class (large smooth area) due to loss of the feature points.

The results also indicate that higher extents of distortion give rise to lower redetection ratio. Nevertheless, as already mentioned, large distortions are not frequently occurring in data hiding applications and hence the proposed elementary triangle extraction appears an appealing method for redetection of the embedded reference information.

5.4 Embedding Reference Information into Elementary Triangles

In this stage, the reference information is parsed into 20 bits binary data and inserted into elementary triangles. The capability of the proposed reference information hiding scheme and the distortions introduced by the process of triangle warping are analyzed.

5.4.1 Experimental Set-Up for Embedding Scheme

The set of extracted feature points is used to divide the image into elementary triangles, and each elementary triangle is warped into a standard triangle. As indicated earlier, the watermark is embedded into the standard triangle and then the standard triangle is unwarped to obtain the watermarked version of the elementary triangle. There are distortions in the embedded watermark information introduced by the process of triangle warping.

During the embedding stage, the standard triangle T_s is warped into the shape of each Delaunay triangle T_D to obtain T_{map}, while each Delaunay triangle T_D is warped back into the shape of the standard triangle so the receiver can extract the reference information from each triangle. Warping of triangles into a standard triangle (for instance an isosceles right triangle) is achieved via affine-transformations.

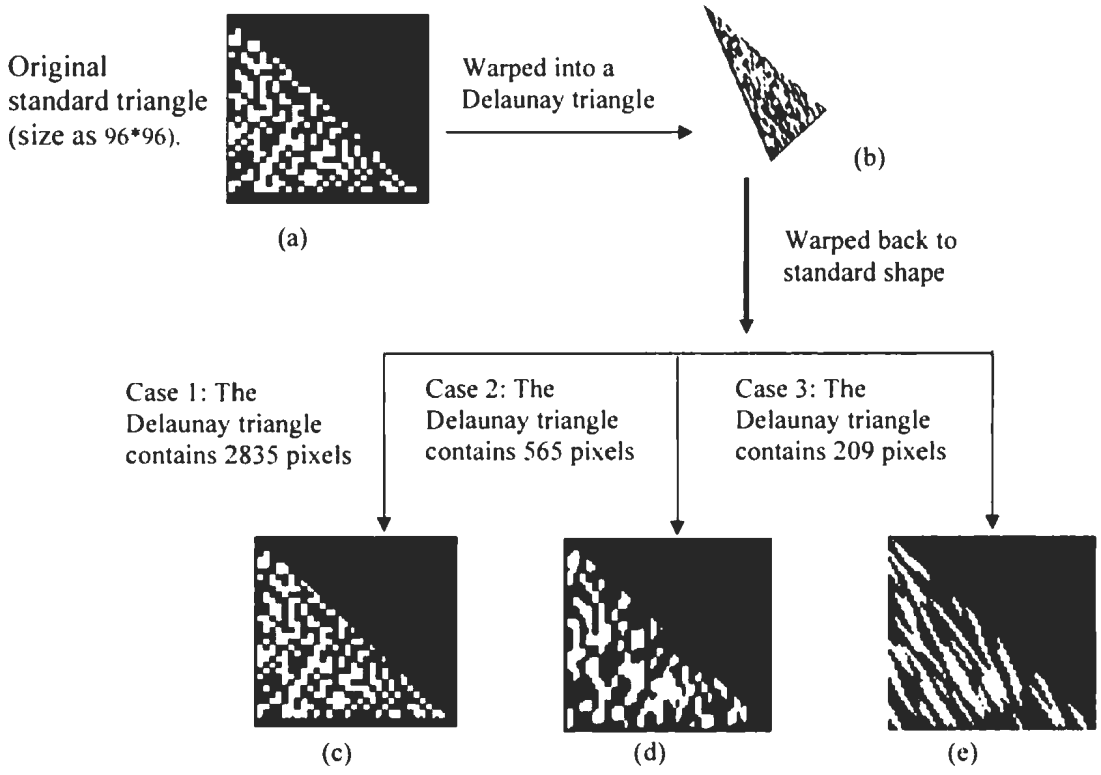


Figure 5.4.1: Triangle warping distortions

The Delaunay triangle size is one of the important factors influencing re-extraction of the embedded reference information. The effect of warping individual Delaunay triangles of different sizes into a standard triangle is given in Figure 5.4.1. Figure 5.4.1(a) shows the original standard triangle and Figure 5.4.1 (c), (d), (e) show the resultant triangles after the transformations of the standard triangle into different sized Delaunay triangles. It is indicated that minimum distortion of the reference information by transformation is obtained when the size of the Delaunay triangle is close to that of the standard triangle.

Spread reference information signals are employed in this experiment, since affine transformations are a blurring process and the embedded information bit may be

lost during this process. Repeated spreading of reference information bits into a block can help recover the lost information bits, as the extracted reference information bit in a block rather than a single bit is determined. In this experiment, the reference information signals are spread on 2*2 pixel blocks.

5.4.2 Evaluation of Data Hiding Scheme

Two classes of geometric distortions, scaling at 10 different scales(0.7, 0.8, 0.85, 0.9, 0.95, 1.05, 1.1, 1.15, 1.2, 1.3) and rotation at 10 different angles (-25, -10, -5, -2, -1, 1, 2, 5, 10, 25) are tested on the image database to evaluate the embedding and extraction scheme. The robustness of the embedding and extraction scheme is measured as the Number of Extracted Reference Information (NERI), which is defined as the number of triangles in which the embedded reference information can be redetected. The number of extracted triangles from the original images and NERI of images after different geometric attacks are presented in Table 5.4.1 and Table 5.4.2. The results suggest that NERI is inversely associated with the distortion extent. Among the image groups, the first class exhibits the highest NERI.

Table 5.4.1: NERI under rotation distortions

	Triangle Number of Original image	NERI after rotation									
		-25	-10	-5	-2	-1	1	2	5	10	25
Lena	48	4	10	15	14	21	23	22	20	10	7
Man	67	6	11	14	21	27	29	21	14	10	9
Couple	59	7	10	13	22	26	27	18	15	11	5
Pentagon	93	3	13	17	28	31	26	24	17	14	2
Baboon	76	0	8	12	14	19	17	16	9	3	1

Boat	63	0	5	13	12	16	15	16	11	4	0
Aerial	98	2	4	9	11	21	18	13	11	3	0
Bridge	85	0	2	7	13	20	21	10	9	2	0
Moon surface	47	3	9	12	14	17	15	16	9	8	4
Hare	45	0	5	9	11	12	14	10	7	4	2
Car	62	2	7	8	11	14	13	12	6	2	0
Peppers	50	3	6	8	11	12	10	8	4	3	2

Table 5.4.2: NERI under scaling distortions

	Triangle Number of Original image	NERI after scaling									
		0.7	0.8	0.85	0.9	0.95	1.05	1.1	1.15	1.2	1.3
Lena	48	5	9	13	17	19	20	17	12	10	8
Man	67	3	11	15	17	21	19	11	7	6	3
Couple	59	4	7	10	17	19	18	14	10	7	3
Pentagon	93	2	5	12	14	16	18	14	11	9	2
Baboon	76	0	4	9	11	16	15	13	6	3	1
Boat	63	0	3	10	9	14	12	11	8	3	0
Aerial	98	0	2	7	9	14	15	10	7	3	0
Bridge	85	0	2	5	9	15	16	5	4	2	0
Moon surface	47	0	4	7	9	12	10	11	4	5	2
Hare	45	0	3	7	9	10	12	8	6	4	2
Car	62	2	7	8	10	13	12	11	5	2	0
Peppers	50	2	4	5	8	10	7	5	2	3	0

5.4.3 Discussions

The rotation angle and scaling factor can be calculated by comparing the extracted reference information and the actual information of the distorted images. In a practical implementation, the specific embedded reference information is usually unknown and the

extractor can not determine if the extracted information bits match the original embedded reference information. In order to identify the embedded reference information, a number of 20 bit binary data sequences are extracted from each individual triangle and these sequences are compared. The presence of two or more identical sequences (NERI ≥ 2) confirms successful extraction of the original embedded information. In this work, the NERI values are higher than 2 within certain ranges of geometric distortions, e.g. the rotation angle from-10° to10° (Table 5.4.1) and the scaling factor from 0.8 to 1.2 (Table 5.4.2).

5.5 Limitations in Practice

This preliminary work provides a possible means of improving the robustness against geometric distortion while keeping a high capacity in the data hiding scheme, and sheds light on embedding multiple bits information through feature-based watermarking. However, limitations exist in practical implementations, one of which is, as discussed earlier, that the geometric distortion estimator is only robust against a certain range of distortions and thus more applicable in data hiding applications. It should be noted also that the images under investigation are restricted to be RGB color images which have more than two color spaces, as required by the embedding scheme.

In addition to rotation and scaling, other distortions such as horizontal and vertical shearing (rescaling the image along X and Y axis to different extents) were also taken into consideration. However, one of the factors inhibiting the shearing distortions from being examined in the proposed scheme is that the loss of redetectable elementary

triangles occurs during horizontal and vertical shearing distortion. More specifically, shearing distortions modify the relative positions of the extracted feature points and consequently alter the tessellation of Delaunay triangles formed; whereas this is not the case for rotation and scaling distortions, in which Delaunay triangulation is independent from the image scales and rotation angles. A good example is that, as shown in Figure 5.5.1, shearing the image along the vertical direction causes varied formation of triangles in the distorted images not corresponding to that in the original image.

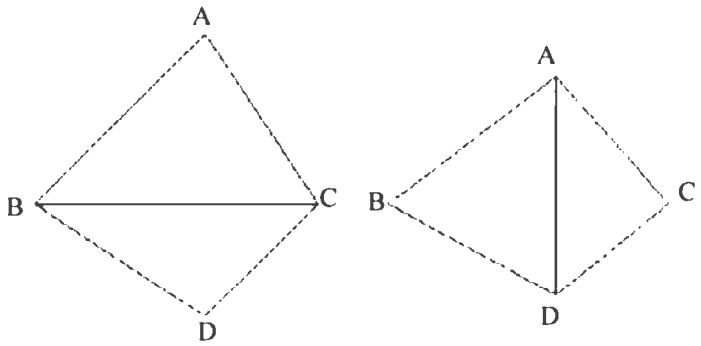


Figure 5.5.1: Effects of aspect ratio change on Delaunay triangulation

Another reason for not considering shearing distortion in this scheme is that introducing one more distortion factor (X and Y shearing factors rather than scaling factor) may increase the information length required, which possibly exceeds the capability of the scheme.

5.6 Contributions and Remarks

This work investigates an image data hiding scheme that counterattacks rotation and scaling distortions. The first contribution of this work is that it presents a method of

embedding geometric reference information into the cover image by a content-based watermarking scheme. This is accomplished by selecting two color spaces of RGB images as two channels (SC and CC channels) to carry geometric reference information of the original image and the information intended to be embedded. In SC channel, a content-based watermarking scheme is used to embed the reference information which can be calculated from the extracted feature points of the original image. The scaling factor and the rotation angle of the image can be estimated and corrected by comparing reference information of the original image and actual information of the transformed image. The capability and robustness of the scheme are evaluated by assessing the three building blocks of the entire scheme. Two or more patches against geometric distortions among the multiple redundant embedding patches in SC channel are required for successful reference information extraction. Simulation results indicate that the embedding and extraction scheme of SC channel show satisfactory performance against certain range of rotation and scaling distortions.

The second contribution of this work is that it provides an estimate of geometric distortion without accessing the original image, which is not available in Masoud's method. It is assumed in Masoud's method that the decoder had prior information regarding the original image maxima, namely, the scale factor and the rotation angle. However, this is not always true since the detector is not always able to locate a specific image from the huge image database. Masoud's method is improved in our work by hiding the prior information about the original image as part of the watermark into the image, so that the prior information remains in the image regardless of distortions.

The third contribution is that it initiates the use of a simplified Harris-Laplace feature point detector instead of standard Harris feature point detector; the former was

suggested according to the results of re-extraction rate a better detection technique with high robustness against scaling distortion by applying the scale space theory.

Another contribution of this work is that it adopts relative reference vertices of Delaunay tessellation triangles to form the elementary embedding triangles. Because the detector is sensitive to any changes of pixels near feature points, relative reference vertices can avoid modification of the sensitive areas and thus improve the stability of the detector. The use of shrunk Delaunay tessellation triangles is found effective in maintaining the redetection ratio of elementary triangles.

Last but not the least, this work explores the possibility of replacing binary data with multiple bits by means of a content-based watermarking scheme for expanded applications. Content-based watermarking scheme is found to be a good option not only for copyright protection and for data hiding as well.

5.7 Directions for Future Work

One important finding is that the feature based embedding scheme proposed in this work, while being robust against geometric distortions, exhibits a capability of more than one bit, suggesting that use of the proposed scheme can be extended to any application requiring short embedding information, such as a serial number of fingerprint. In the fingerprint applications, the detector must successfully extract the embedded sequence numbers in order to track the user who leak the fingerprint. The scheme proposed in this work is of satisfaction at this point, implicating its use in fingerprint applications alike.

Moreover, the scheme can be integrated into any existing data hiding schemes for RGB images, since only one channel of RGB image is needed, hence improving their robustness against geometric distortions.

As already discussed, limitations of the data hiding scheme include the limited range of distortions in which accuracy of geometric distortion estimation is guaranteed. Further investigation is required to achieve an extended distortion range and thus enhance the effectiveness of the geometric distortion estimator. The effectiveness of the estimator can be further enhanced by increasing the robustness of feature point detection, which is positively correlated with the redetection ratio of elementary triangles. Searching for more stable feature points and more reliable extraction algorithms under severe geometric distortions is a necessity. More studies on scale space theory are recommended for improvement of redetection ratio of feature points.

With respect to the accuracy of geometric distortion estimation, possible coarse estimation of scaling and rotation can be minimized and even eliminated using an exhaustive search method, i.e. a hierarchical search, using estimated scale factors and rotation factors to localize the range of the search and to refine the estimate by searching around these values.

Meanwhile, further research on enhancing the embedding capability of the scheme can be carried out, e.g. by introducing more powerful error correcting codes in the embedding and extraction scheme. Although the combination of convolutional encoding and soft-decision Viterbi decoding is the most commonly used error-correcting coding in content-based watermarking studies, as used in this work, other techniques can be investigated for an optimized error correcting method. Similarly, possibilities of using other noise removal filters than the Wiener filter can be explored.

Bibliography

1. Brassil, J., S. Low, and N.F. Maxemchuk, *Copyright protection for the electronic distribution of text documents.* Porceedings of the IEEE, Special Issue on Protection of Multimedia Content, July, 1999. **87**: p. 1181-1196.
2. Hattacharijya, A.K.B. and H. Ancin, *Data embedding in text for a copier system.* In Proceedings of the 6th IEEE International Conference on Image Processing, October 1999. **2**: p. 245-249.
3. Nikolaidis, N. and I. Pitas, *Digital image watermarking: An overview.* In Proceedings of the IEEE International Conference on Multimedia and Computing Systems, June 1999. **1**: p. 1-6.
4. Linnartz, J.-P., T. Kalker, and J. Haitsma, *Detecting electronic watermarks in digital video.* In proceedings of ICASSP '99, March 1999. **4**: p. 2071-2074.
5. Qiao, I. and K. Nahrstdt, *Watermarking methods for MPEG encoded video.* In Proceedings of IEEE International Conference on Multimedia Computing and Systems, 1998: p. 276-285.
6. Lu, C.-S. and H.-Y.M. Liao, *Multipurpose audio watermarking.* In proceedings of the 15th International Conference on Pattern Recogition September 2000. **3**: p. 3286.
7. Li, X. and H. Heather, *Transparent and robust audio data hiding in subband domain.* In Proceedings of the IEEE International Conference on Information Technology: Coding and Computing, March 2000: p. 74.
8. Ohbuchi, R., H. Masuda, and M. Aono, *Watermarking three-dimensional polygonal models through geometric and topological modificaitons.* In IEEE Journal on Special Areas in Communications May 1998. **16**: p. 551-560.
9. Masoud, A. and T. Ahmed, *Geometric distortion correction in image watermarking.* Proc. SPIE 2000. **3971**: p. 82-89.
10. Xue, G. and P. Lu, *A counter-geometric distortions data hiding schemes using double channels in color images.* IWDW, 2004: p. 42-54.
11. Potar, V.M., S. Han, and E. Chang, *A survey of Digital Image Watermarking Techniques.* 3rd IEEE International Conference on Industrial Informatics, 2005: p. 709-716.
12. Sequeira, A. and D. Kundur, *Communication and information theory in watermarking : a survey.* Proc. SPIE Multimedia Systems and Applications IV, 2001. **4518**: p. 216-227.
13. Furon, T., *A survey of watermarking security.* Proc. of Int. Work. on Digital Watermarking, ed. M.Barni. Vol. 3710. 2005: Springer-Verlag. 201-215.
14. George, M., J.-Y. Chouinard, and N. Georganas, *Spread Spectrum Spatial and Spectral Watermarking for Images and Video using Direct Sequence Techniques*, in *1999 IEEE Canadian Workshop on Information Theory*. 1999. p. 119-122.
15. Langelaar, G., I. Setyawan, and R. Lagendijk, *Watermarking digital image and video data.* IEEE Signal Processing Magazine, 2000. **17**: p. 20-46.
16. Cox, I.J., et al., *Secure spread spectrum watermarking for multimedia.* IEEE Trans. on Image Processing, 1997. **6**(12): p. 1673-1687.
17. Marvel, L., J. CG Boncelet, and C. Retter, *Spread spectrum image steganography.* IEEE Transactions on image processing, August, 1999: p. 1075-1083.

18. Mayer, J., A.V. Siverio, and J.C.M. Bermudez, *On the design of pattern sequences for spread spectrum image watermarking* in *IEEE Int. Telecommunications Symposium (ITS'02)*. 2002: Natal,Brazil.

19. George, M., J.-Y. Chouinard, and N. Georganas, *Digital Watermarking of images and video using direct sequence spread spectrum techniques.* IEEE Can. Conf. Electrical Comput. Eng., 1999. **1**(9-12): p. 116-121.

20. Hernandez, J., J. Delaigle, and B. Macq, *Improving Data Hiding by Using Convolutional Codes and Soft-Decision Decoding.* Proceedings of SPIE on Security and Watermarking of Multimmedia 2000. **3971**: p. 24-47.

21. Su, J.K., F. Hartung, and B. Girod, *Digital watermarking of text, image and video documents.* Computer and Graphics, 1998. **22**(6): p. 687-695.

22. Oostveen, J., T. Kalker, and J.P. Linnartz, *Optimal detection of multiplicative watermarks.* European Signal Processing Conference, 2000. **5**: p. 2973-2976.

23. Barni, M., et al., *A m.a.p. identification criterion for DCT-based watermarking.* Proc.Europ. Signal Processing Conf. (EUSIPCO 98), 1998. **1**: p. 17-20.

24. Johnson, N.F. and S.C. Katezenbeisser, *A survey of steganographic techniques.* Information Techniques for steganography and Digital Watermarking, 1999: p. 43-75.

25. Costa, M.H.M., *Writing on dirty paper.* IEEE Trans. on Information Theory, 1983. **IT-29**(3): p. 439-441.

26. Chen, B. and G.W. Wornell, *Quantization Index Modulation Methods: A Class of Provably Good Methods for Digital Watermarking and Information Embedding.* IEEE Trans. on Information Theory, 2001. **47**: p. 1423-1443.

27. Eggers, J.J., R. Bauml, and R. Tzschoppe, *Scalar Costa scheme for information embedding.* IEEE Trans. on Signal Processing 2003. **51**(4): p. 1003-1019.

28. Solanki, K., et al. *High-volume data hiding in images: Introducing perceptual criteria into quantization based embedding.* in *Proceedings of ICASSP*. 2002.

29. Meerwald, P., *Quantization watermarking in JPEG 2000 coding pipleline.* 5th International Working Conference on Communication and Multimedia Security, 2001. **192**: p. 19.

30. Bender, W., et al., *Techniques for Data Hiding.* IBM Systems Journal, 1996. **35**: p. 313-336.

31. Fridrich, J. *Robust bit extraction from images.* in *IEEE International Conference on Multimedia Computing and Systems*. 1999.

32. Wolfgang, R.B. and E.J. Delp, *A watermark for digital images.* Proceedings of International Conference on Image Processing, 1996. **3**: p. 219-222.

33. Koch, E. and J. Zhao, *Towards Robust and hidden image copyright labeling.* IEEE Workshop on Nonlinear Signal and Image Processing, 1995: p. 452-455.

34. Noore, A., *An improved digital watermarking technique for protecting JPEG images* IEEE International Conference on consumer electronics, 2003: p. 222-223.

35. Fotopoulos, V. and A.N. Skodras. *A Subband DCT Approach to Image Watermarking.* in *Proceedings of X European Signal Processing Conference*. 2000. Tampere, Finland.

36. Suhail, M. and A. Obaidat, *Digital Watermarking Based DCT and JPEG Model.* IEEE Trans. on Instrumentation and Measurement, 2003. **52**(5): p. 1640-1647.

37. Choi, Y. and K. Aizawa, *Digital Watermarking Techniques using Block Correlation of DCT Coefficients.* IEICE Transactions on Information and Systems, PT.2, 2002. **J83-D-2**(7): p. 1620-1627.

38. Huang, J., Shi,YQ and Y. Shi, *Embedding Image Watermarks in DC Components.* IEEE Trans. on circuit and system for video technology, 2000. **10**(6): p. 974-979.

39. Hsu, C.-T. and J.-L. Wu, *Hidden Digital Watermarks in Images.* IEEE Trans. on Image Processing, 1999. **8**(1): p. 56-58.

40. Tao, B. and B. Dickinson, *Adaptive Watermarking in DCT Domain.* IEEE International Conference on Acoustics, Speech, and Signal Processing, 1997. **4**: p. 1985-2988.

41. Lu, C.S., H.-Y. Liao, and M.Huang, *Cocktail Watermarking on Images*, in *3rd International Workshop on Information Hiding,*. 1999: Dresden, Germany.

42. Zhu, W., Z. Xiong, and Y.Q. Zhang, *Multiresolution Watermarking for Images and Videos.* IEEE Trans. on circuit and system for video technology, 1999. **9**(4): p. 545-550.

43. Voyatzis, G. and I. Pitas, *Digital Image Watermarking using Mixing Systems.* Computer and Graphics, 1998. **22**(4): p. 405-416.

44. Kundur, D. and D. Hatzinakos, *Digital Watermarking using Multiresolution Wavelet Decomposition.* IEEE Int. Conf. On Acoustics, Speech and Signal Processing, 1998. **5**: p. 2969-2972.

45. Xia, X.-G., C. G.Boncelet, and G. R.Arce, *Wavelet transform based watermark for digital images* Optics Express, December 1998. **3**(12): p. 497.

46. Lu, C.S., S.K.Huang, and H.-Y. Liao, *A new watermarking techniques for multimedia protection.* Multimedia image and video processing, 2001: p. 507-530.

47. Meerwald, P. and A. Uhl, *A survey of wavelet-domain watermarking algorithms.* In proceedings of SPIE, Security and Watermarking of Multimedia Contents III, January 2001. **4314**: p. 505-516.

48. Feig, E., *A fast scaled DCT algorithm.* Proc. SPIE Image Processing Algorithms and Techniques, 1990. **1224**: p. 2-13.

49. O'Ruanaidh, J.J.K. and T. Pun, *Rotation, scale and translation invariant digital image watermarking.* Image processing, international conference, 1997. **1**: p. 536-539.

50. O'Ruanaidh, J.J.K., W.J. Dowling, and F.M. Borland, *Phase watermarking of digital images.* Proc. IEEE Int. Conf. Image Processing, 1996: p. 239-242.

51. Pereira, S. and T. Pun, *Robust Template Matching for Affine Resistance Image Watermarks.* IEEE Trans. on Image Processing, 2000. **9**(6): p. 1123-1129.

52. Tang, C.-W. and H.-M. Hang, *A feature-based robust digital image watermarking scheme.* Signal Processing, IEEE Transactions 2003. **51**(4): p. 950-959.

53. Bas, P.C., J.-M. Macq, B., *Geometrically invariant watermarking using feature points.* Image Processing, IEEE Transactions, Sep 2002. **11**(9): p. 1014-1028.

54. Doerr, G. and J.L.Dugelay, *Security pitfalls of frame-by-frame approaches to vidoe watermarking.* IEEE Trans.Sig.Proc., Supplement on Secure Media 2004. **52**: p. 2955-2964.

55. Fridrich, J. and M. Goljan, *Robust hash functions for digital watermarking* Proceedings of the International Conference on Information Technology: Coding and Computing, 2000: p. 173-178.

56. Voloshynovskiy, S., et al., *A stochastic approach to content adaptive digital image watermarking* Third Workshop on Information Hiding, Lecture Notes in Computer Science, 1999. **1768**: p. 211-236.

57. Su, J., J. Eggers, and B. Girod, *Analysis of digital watermarks subjected to optimum linear filtering and additive noise.* Signal Processing, 2001. **81**: p. 1131-1175.

58. Pateux, S., G.L. Guelvouit, and C. Guillemot, *Perceptual watermarking of non i.i.d signals based on wide spread spectrum using side information* Proceedings of the IEEE International Conference on Image Processing, 2002. **3**: p. III-477- III-480.

59. Linnartz, J.P. and M.v. Dijk, *Analysis of the sensitivity attack against electronic watermarks in images.* Proc. 2nd Int. Workshop Information Hiding,, 1998. **1525**: p. 258-272.

60. Comesana, P. and L. Perez-Freire, *The return of the sensitivity attack.* Proc. 4th Int. Workshop Digital Watermarking., 2005. **3710**: p. 260-274.

61. Mansour, M.F. and A.H. Tewfik, *LMS-based attack on watermark public detectors.* 2002. **3**: p. 649-652.

62. Licks, V. and R. Jordan, *Geometric attacks on image watermarking systems.* IEEE Multimedia, 2005. **12**(3): p. 68-78.

63. Nikolaidis, N. and I. Pitas, *Copyright protection of images using robust digital signatures.* Proc. IEEE Int. Conf. Acoustics,Speech, Signal Processing, May, 1996. **4**: p. 2168-2171.

64. Xia, X.G., C.G. Boncelet, and G.R. Arc, *A multiresolution watermark for digital images.* Proc. ICIP'97, Oct. 1997. **1**: p. 548-551.

65. Petitcolas, A.P., J.A. Ross, and G.K. Markus, *Attacks on copyright marking systems.* Information hiding: second international workshop, 1998. **1525**: p. 218-238.

66. Kutter, M., S.K. Bhattacharjee, and T. Ebrahimi, *Towards second generation watermarking schemes.* Image Prcoessing, 1999.ICIP99. Proceedings. , 1999. **1**: p. 320-323.

67. Pereia, S. and T. Pun, *Fast robust template matching for affine resistant image watermarking.* Lecture Notes in Computer Scinece, Sept.29-Oct.1,1999. **LNCS 1768**: p. 200-210.

68. Fleet, D.J. and D.J. Heger, *Embedding invisible information in color images.* Proc. IEEE Int'l Conf. Image Processing 1997. **1**: p. 532-535.

69. Chen, B. and G.W. Wornell, *An information-theoretic approach to the design of robust digital watermarking systems.* Proceedings of the Acoustics, Speech, and Signal Processing, , 1999. **4**: p. 2061-2064.

70. Kutter, M., *Watermarking Resisting to Translation, Rotation, and Scaling.* Proc.SPIE Multimedia Systems and Applications, 1999. **3528**: p. 423-554.

71. Hartung, F., J.K. Su, and B. Girod, *Spread Spectrum Watermarking: Malicious Attacks and Counterattacks.* Proc. SPIE: Security and Watermarking of Multimedia Contents, 1999. **3657**: p. 147-158.

72. Lin, C.-Y., et al., *Rotation, Scale, and Translation Resilient Watermark for images.* Image Processing, IEEE Transactions May 2001. **10**(5): p. 767-782.

73. Alghoniemy, M. and A.H. Tewfik, *Geometric distortion correction through image normalization.* Proc. Int. Conf. Multimedia Expo., 2000. **3**: p. 1291-1294.

74. Brandt, R.D. and F. Lin, *Representations that uniquely characterize images modulo translation, rotation, and scaling.* Pattern Recognition Lett., 1996. **17**(9): p. 1001-1015.

75. Voloshynovskiy, S., F. Deguillaume, and T. Pun, *Multibit digital watermarking robust against local nonolinear geometrical distortions.* PROC. ieee Int. Conf. Image processing, , 2001: p. 999-1002.

76. Seo, J.S. and C.D. Yoo, *Localized image watermarking based on feature points of scale-space representation.* Pattern Recognition, 2004. **37**: p. 1369-1375.

77. Nikolaidis, A. and I. Pitas, *Region-based image watermarking.* IEEE Trans. on Image Processing, 2001. **10**: p. 1726-1740.

78. Rongen, P.M.J., M.J. Maes, and C.W.v. Overfeldq, *Digital Image Watermarking by Salient Point Modification Practial Results.* Proc. SPIE 1999. **3657**: p. 273-282.

79. Celik, M., et al., *Analysis of Feature-based Geometry Invariant Watermarking.* Proc. SPIE:Security and Watermarking of Multimedia Contents III, 2001. **4314**: p. 261-268.

80. Schmid, C. and R. Mohr, *Local grayvalue invariants for image retrieval.* Pattern analysis and machine intelligence, IEEE transactions, 1997. **19**(5): p. 530-535.

81. Harris, C. and M. Stephen, *A combined corner and edge detector.* Proc.of the 4th Alvey Vision Conf, 1988: p. 147-151.

82. Schmid, C., R. Mohr, and C. Bauckhage, *Evaluation of interest point detectors.* IJCV, 2000. **37**(2): p. 151-172.

83. Mikolajczyk, K. and C. Schmid, *Indexing based on scale invariant interest points.* ICCV, 2001: p. 525-531.

84. Gracias, N. and J. Santos-Victor, *Underwater video mosaics as visual navigation maps.* Computer vision and image understanding, July, 2000. **79**(1): p. 66-91.

85. Lee, J.S., *Digital image enhancement and noise filtering by use of local statistics.* IEEE TRANS. Pattern Anal. Machine Intell, 1980. **PAMI-2**: p. 164-168.

Appendix A: Testing Image Database

Category 1:

Lena

Man

Couple

Pentagon

Category 2:

Baboon

Boat

Aerial

Stream and bridge

Category 3:

Moon surface

Hare

Car

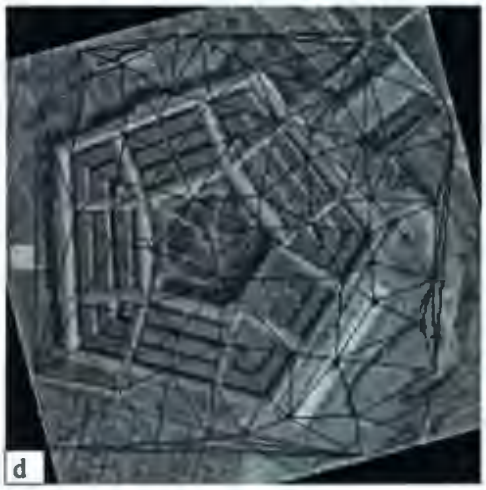
Peppers

Appendix B: Triangle Redetection under Geometrical distortions

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
48	38	29	23	12

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
67	49	39	22	28

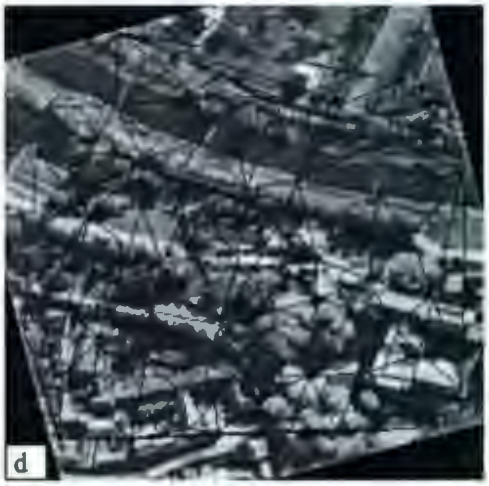
(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
74	49	43	21	18



(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c) Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
73	40	37	19	14

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c) Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
56	34	27	18	11

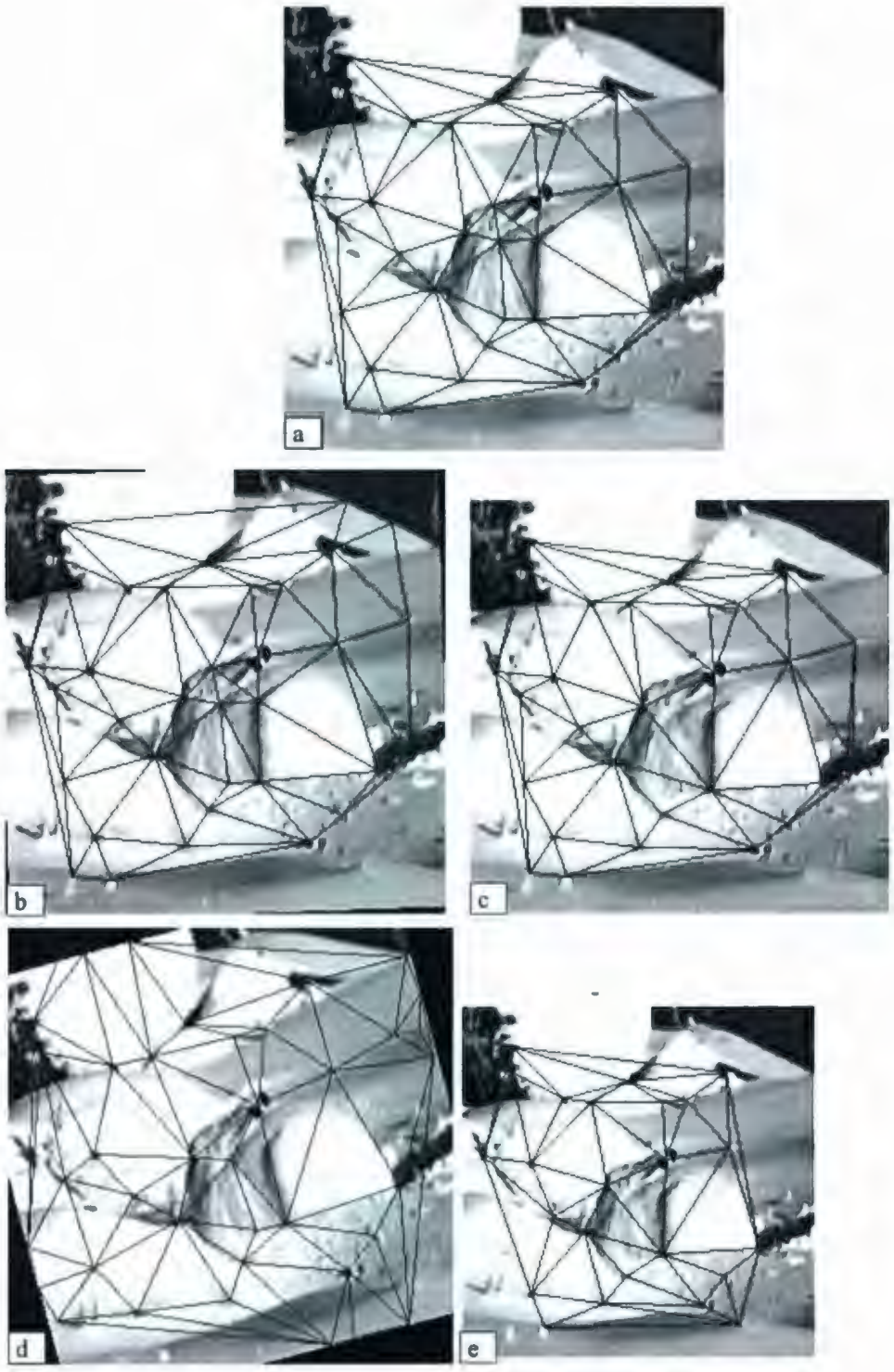
(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
70	41	36	13	8



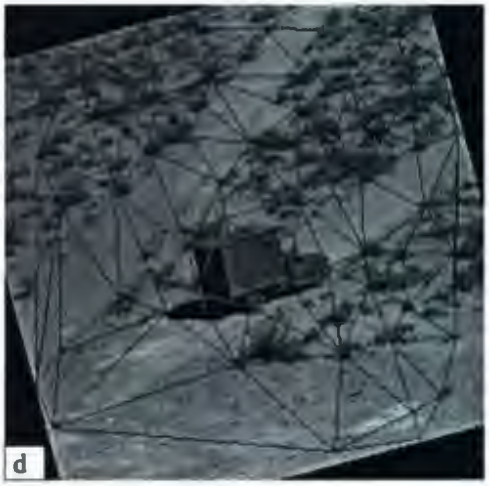
(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
74	45	38	17	13

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
79	48	36	16	10

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
47	29	22	12	7



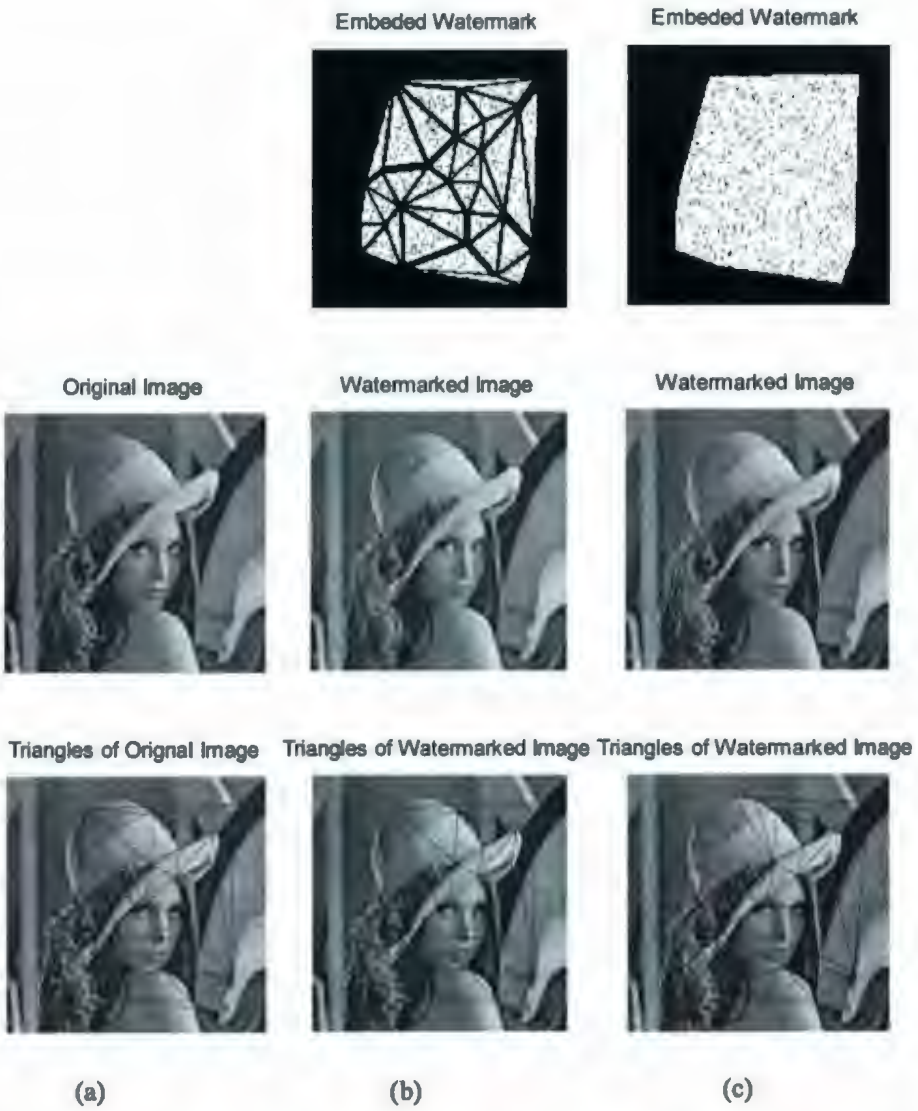
(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
51	30	24	16	9



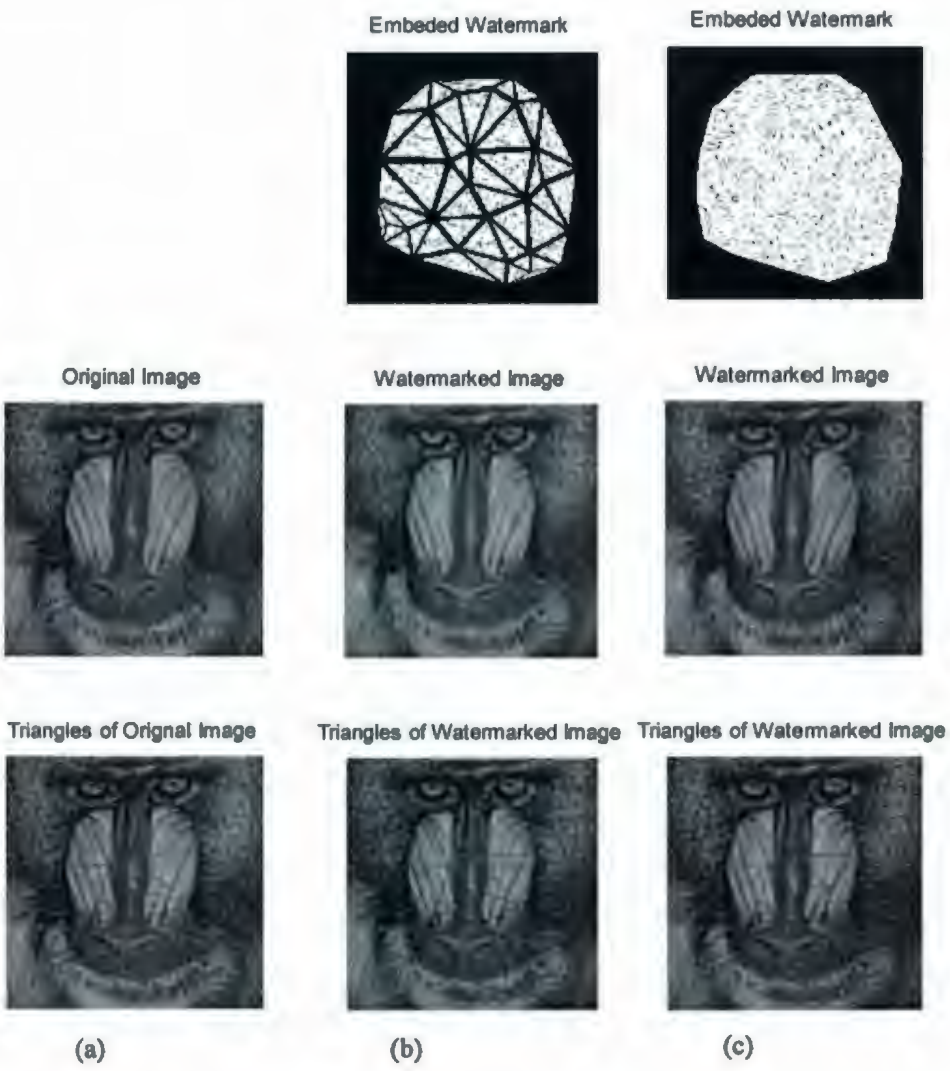
(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
72	39	27	18	8

(a) Original Triangles	Redetected triangles (identical to the original triangles) under distortions			
	(b) Rotating 2°	(c)Scaling 0.9	(d) Rotating 15°	(e) Scaling 0.75
44	23	20	18	7

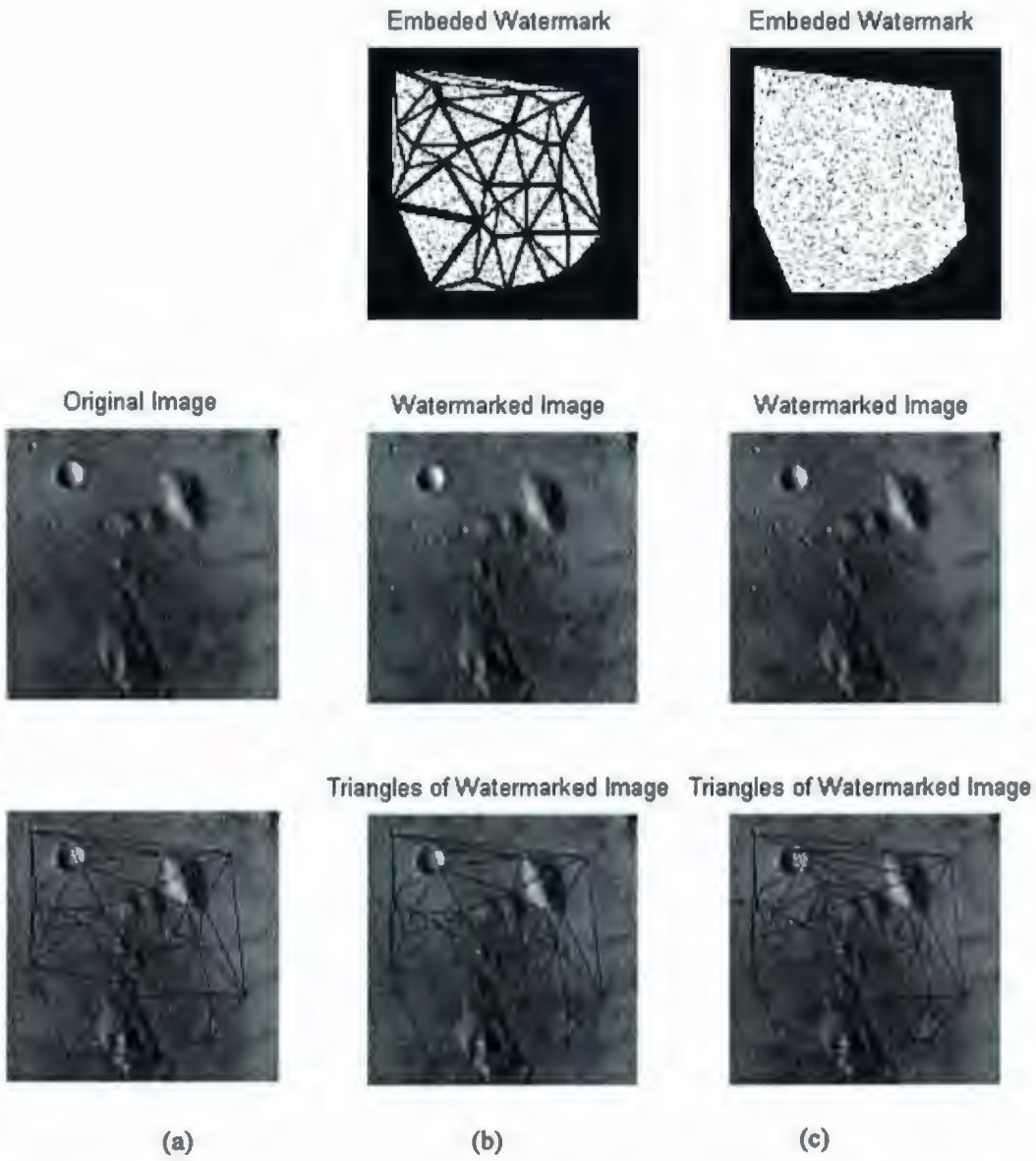
Appendix C: Modification of Delaunay Triangle



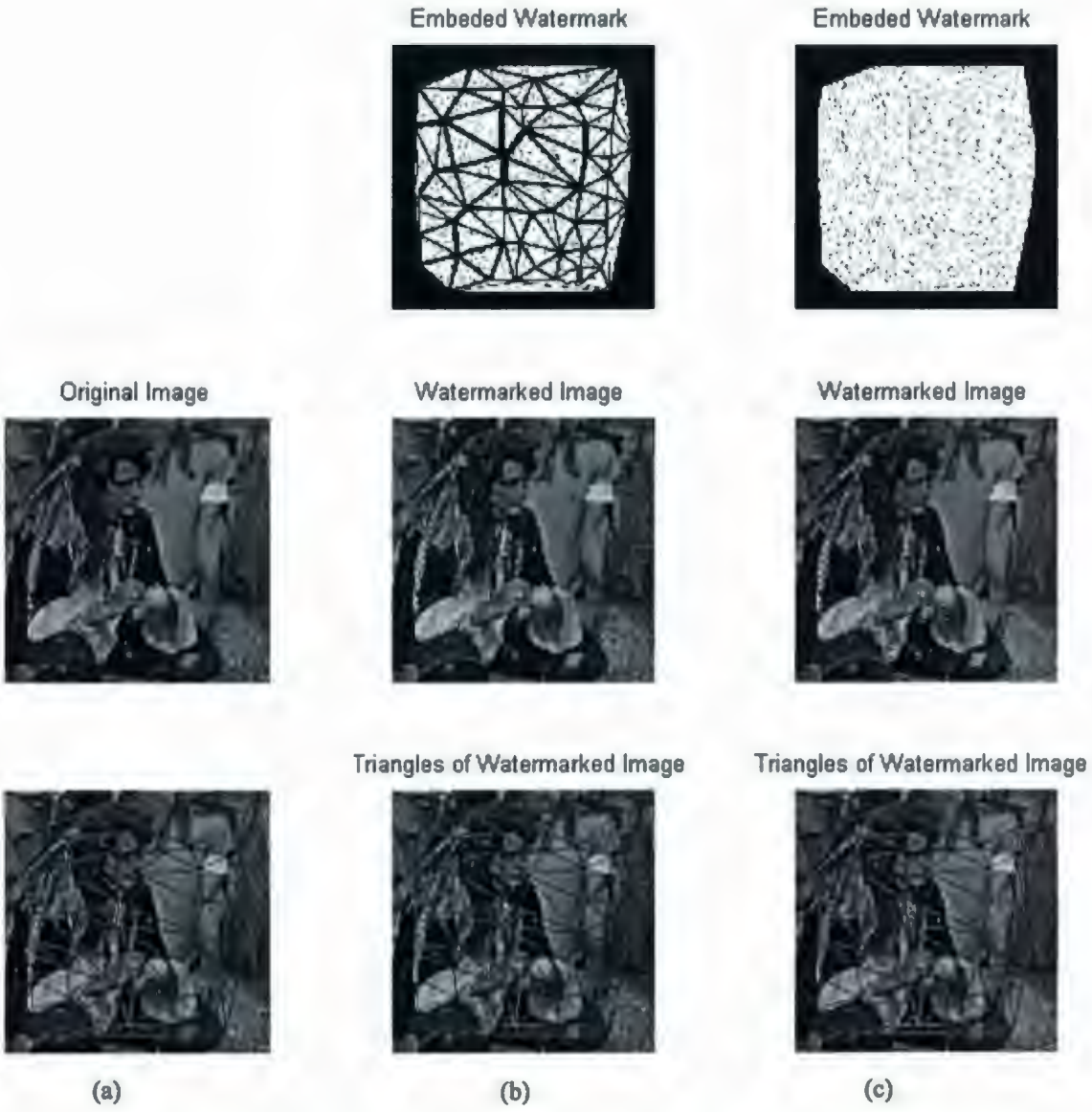
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



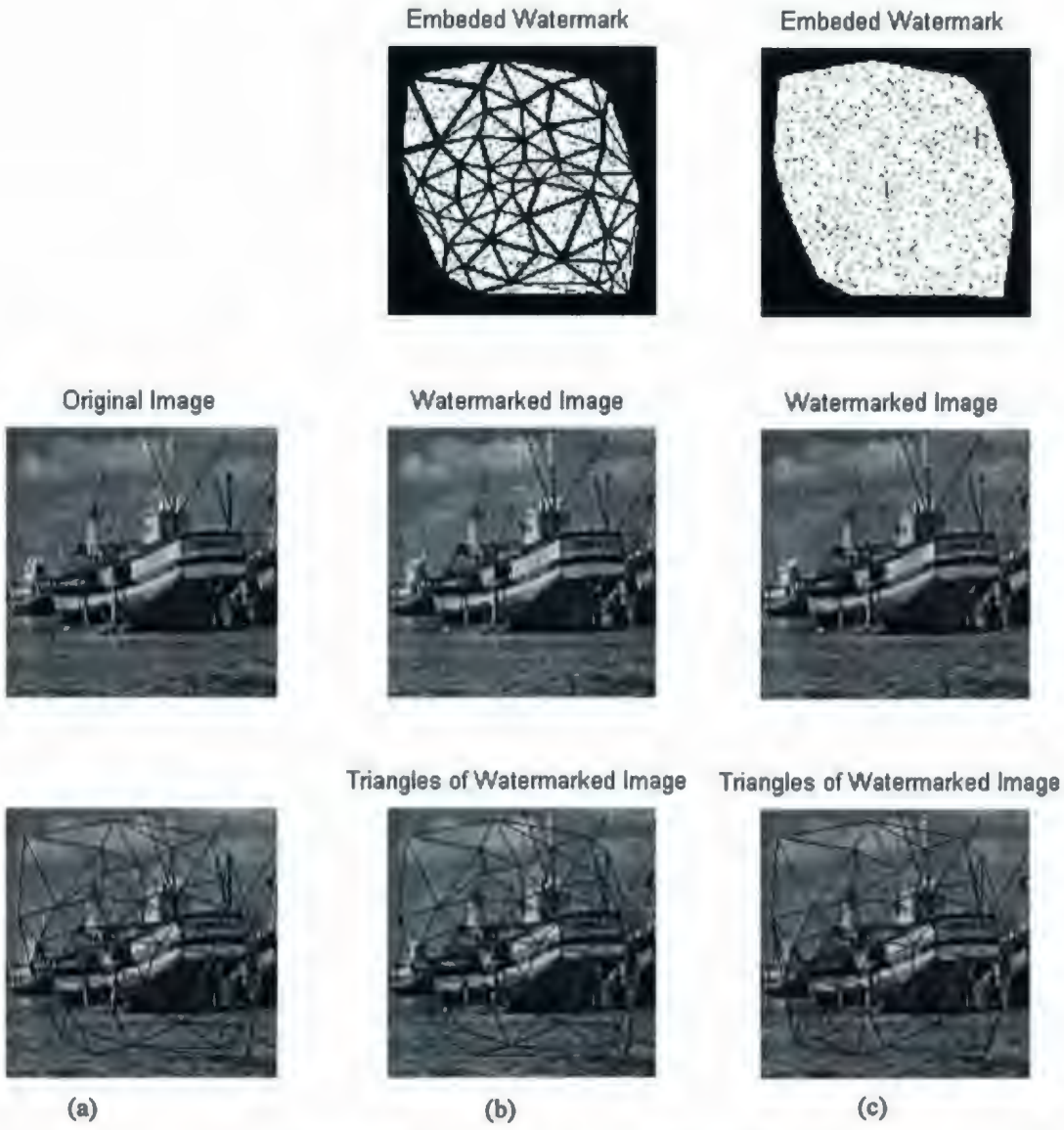
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



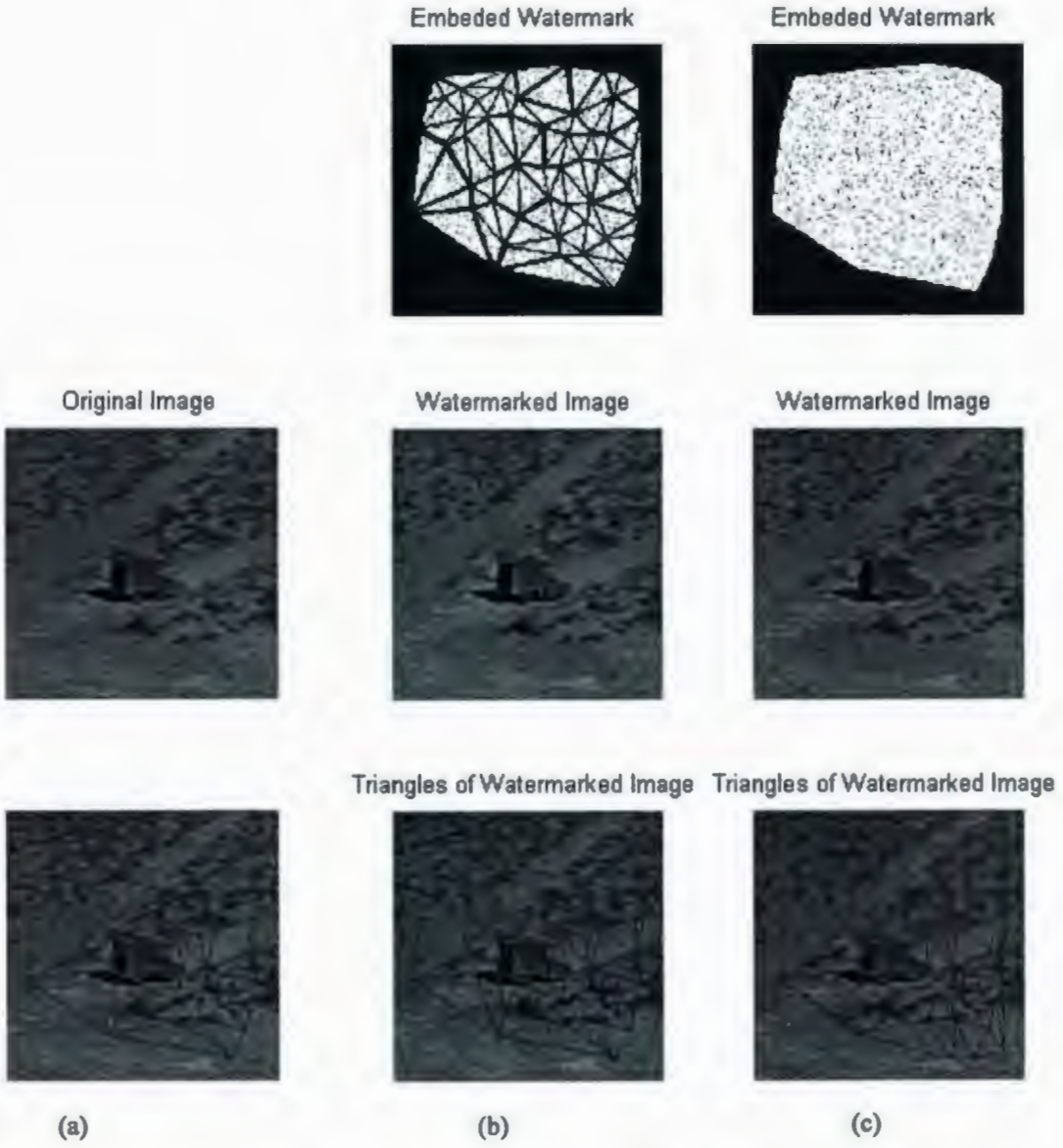
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



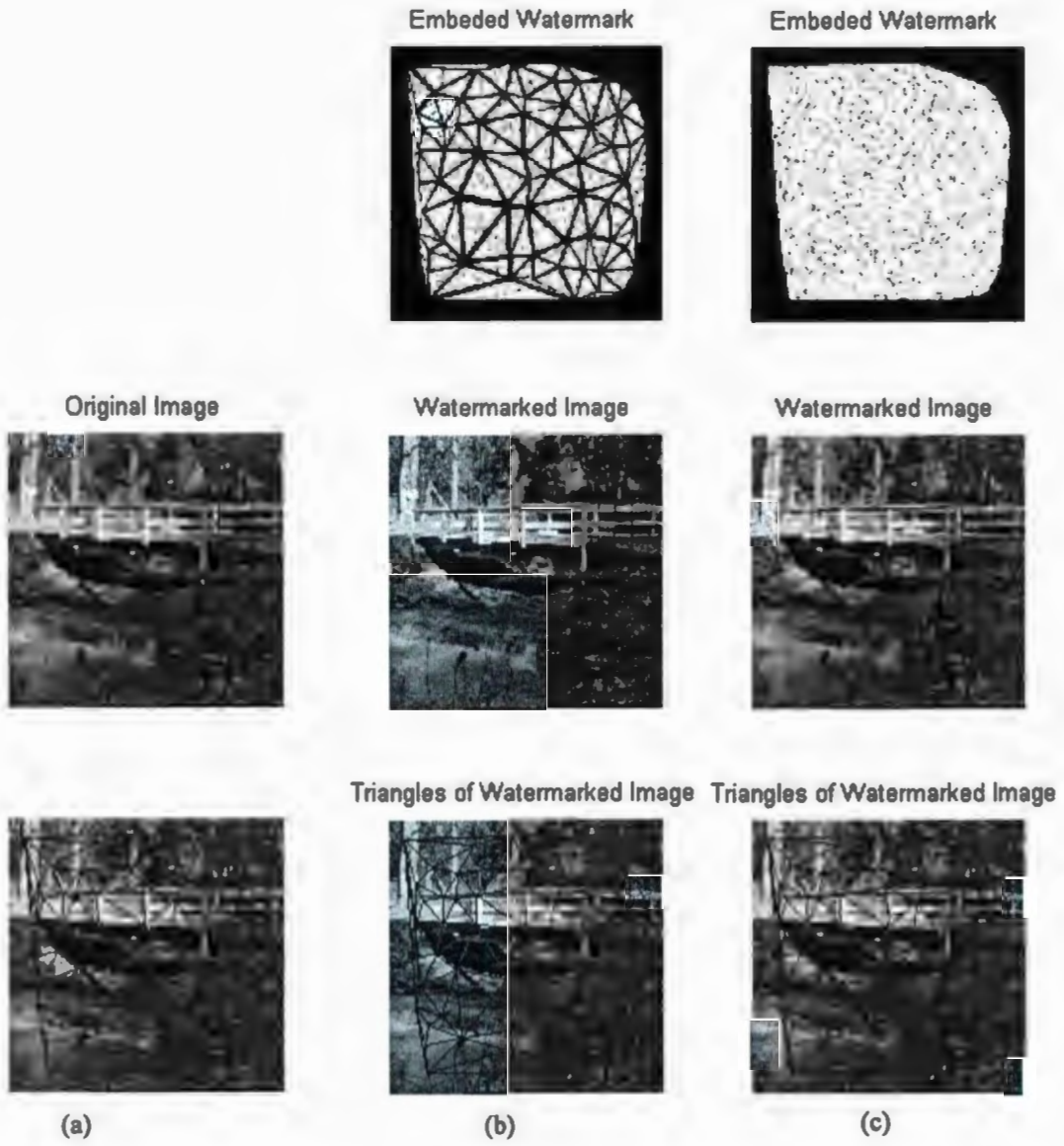
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



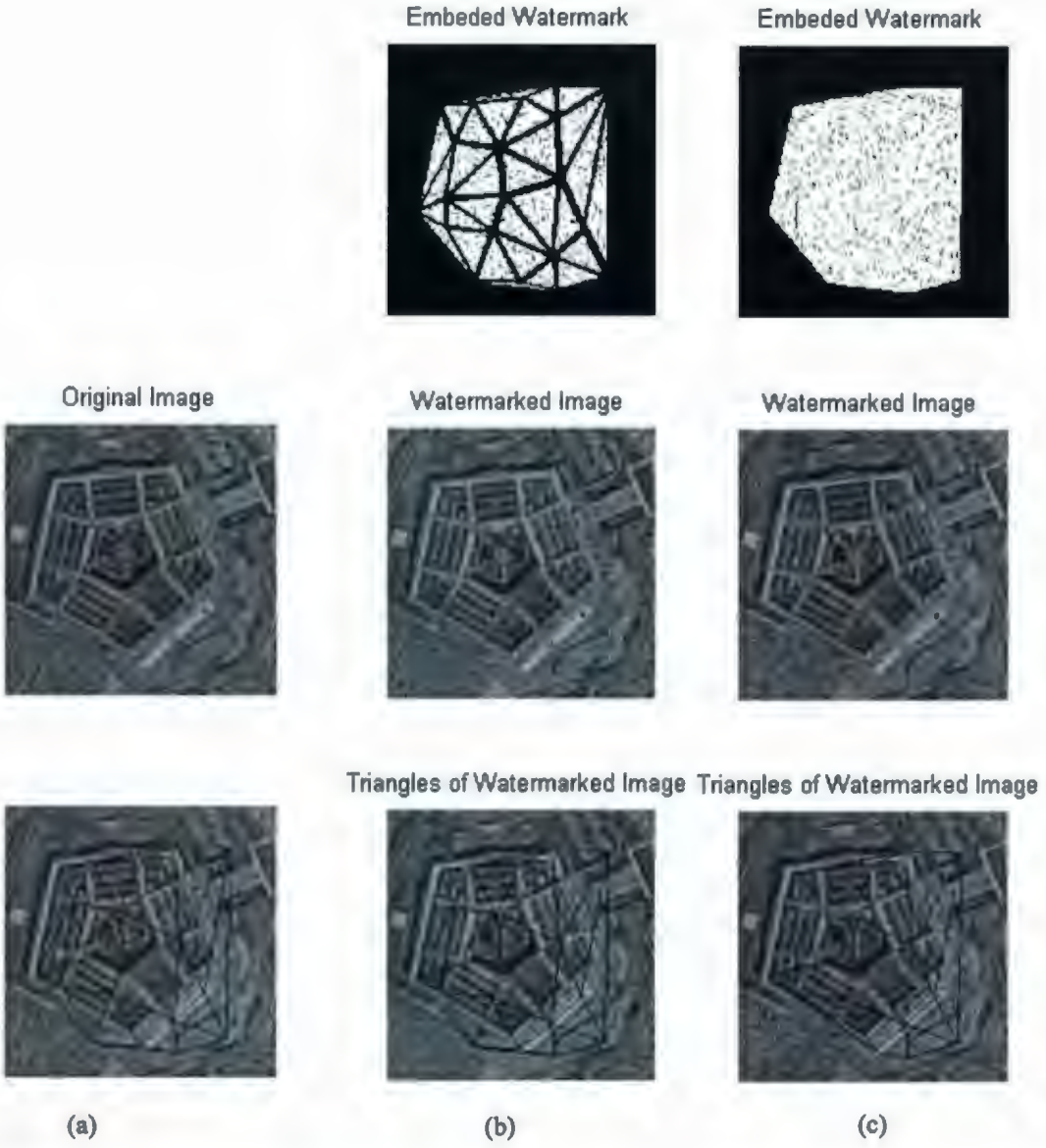
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



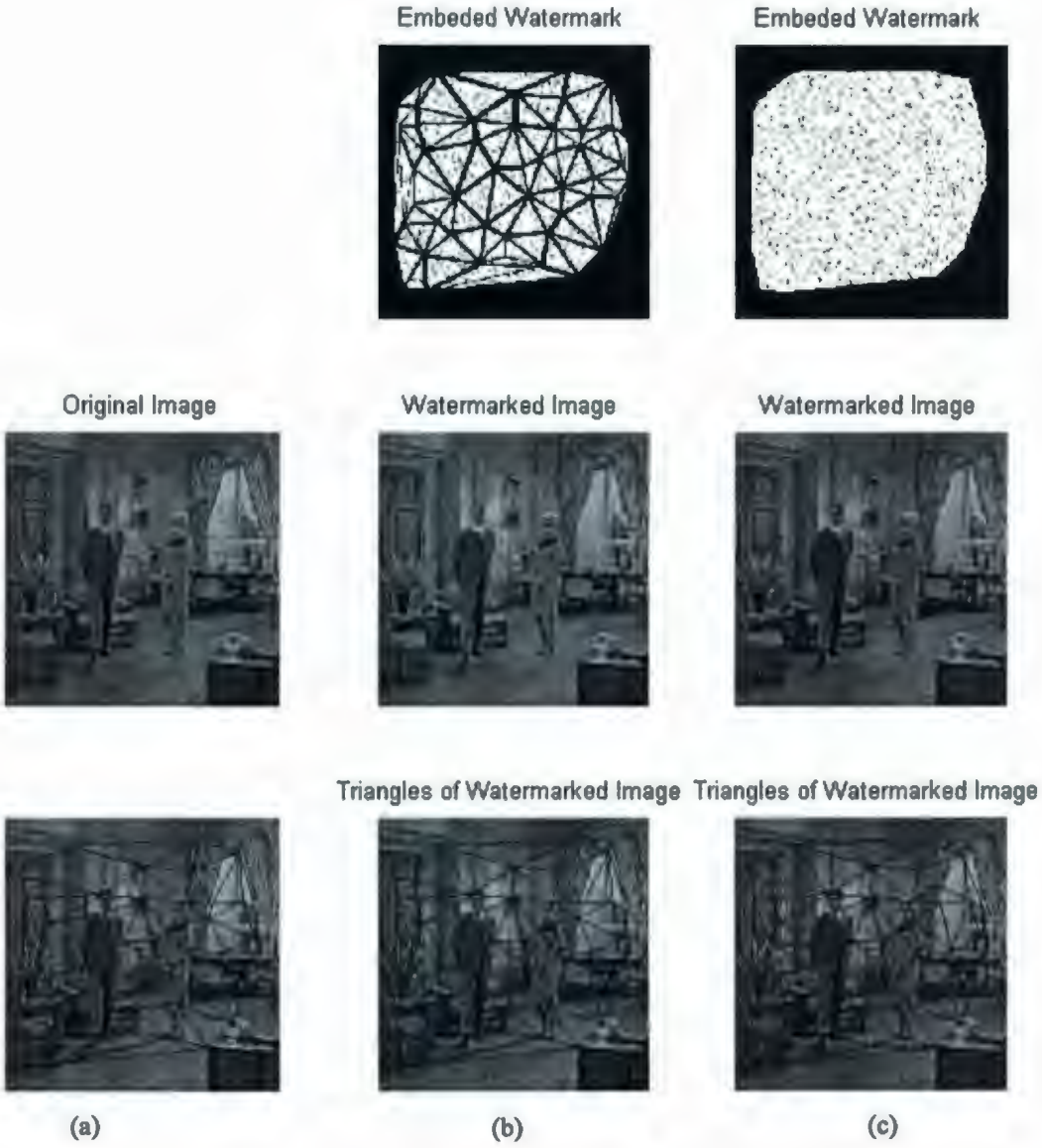
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



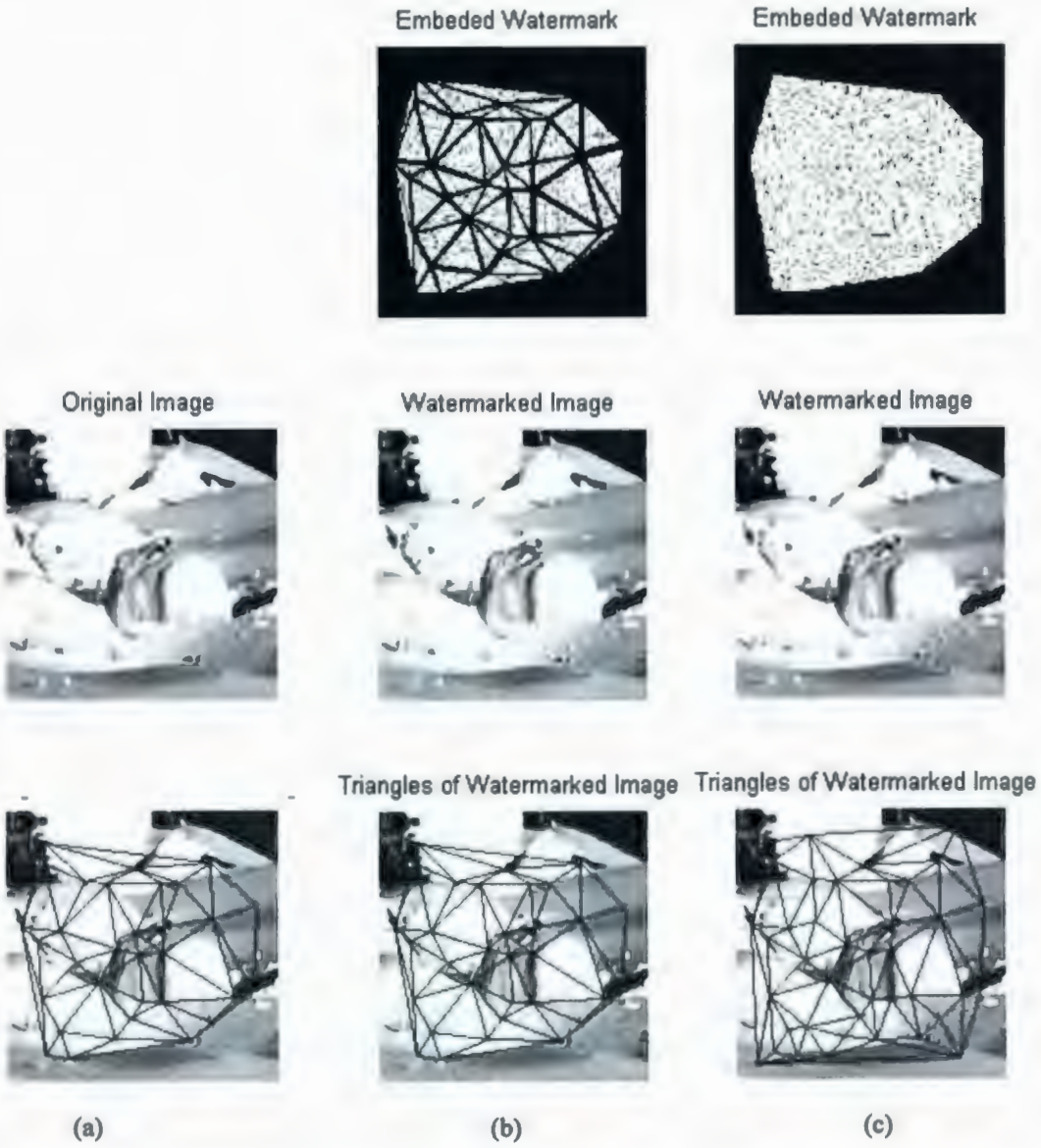
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



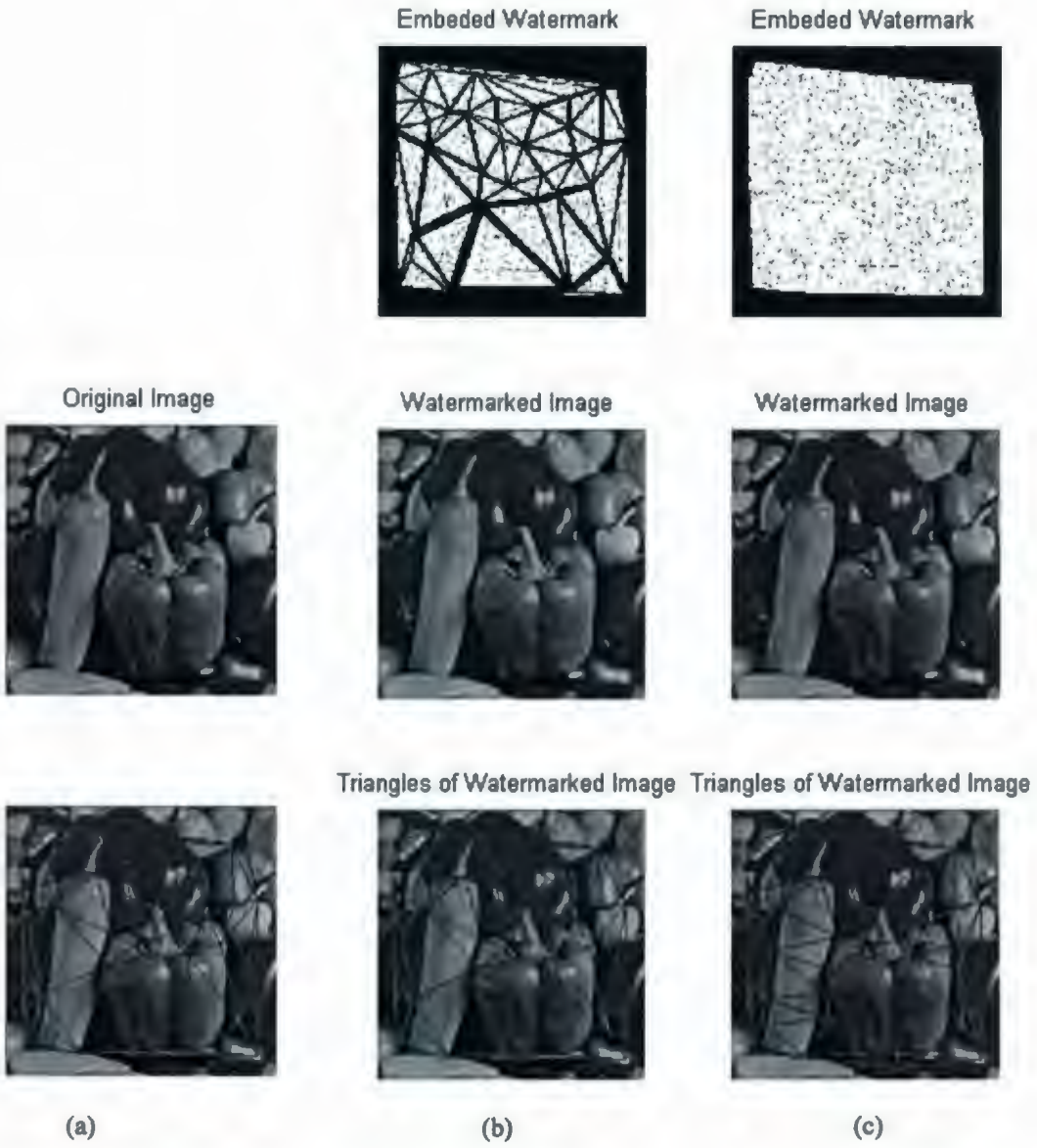
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



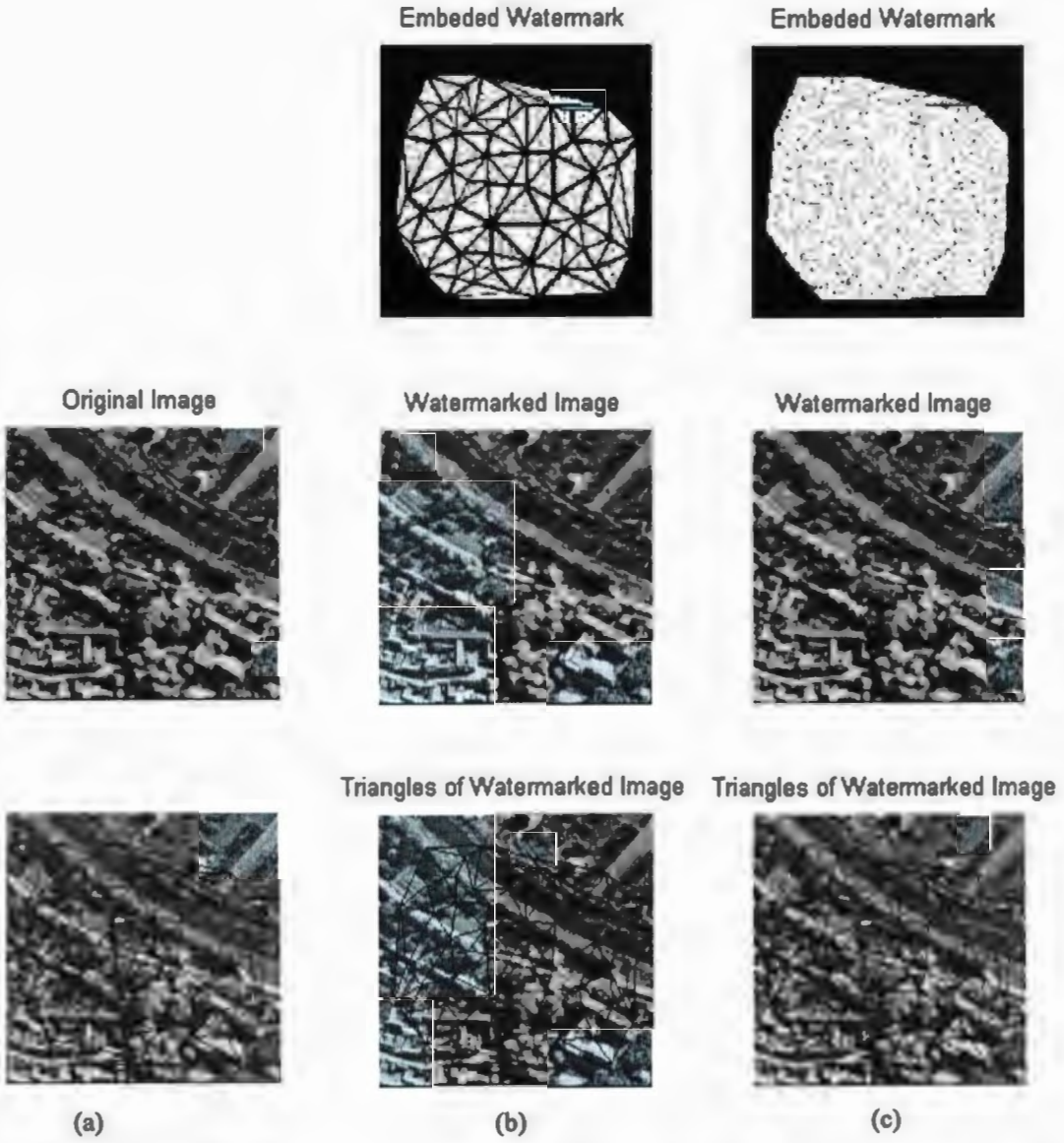
Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)



Modification of the Delaunay triangles: (a) extract triangle from original image (b) re-extract triangle from watermarked image (embedding with modified Delaunay triangles) (c) re-extract triangle from watermarked image (embedding with no-modified Delaunay triangles)